SIMES - 961620
Système d’Information Multimédia
Pour l’Environnement Subsaharien

Modelling

Deliverable number : D11
Nature:P
Contractual Date of Delivery: june, 9, 2000

Task WP2.1 : Data modelling

Nom du rédacteur : Olivier Monga
Institut : Inria
E-mail : Olivier.Monga@inria.fr

Abstract:

Ce document présente l'utilisation des méthodes les plus récentes de traitement de l'information (traitement d'images, analyse statistique, représentation symbolique) pour l'analyse des données du projet SIMES (voir Workpackage 1: "Acquisition de données"). La validation de la méthodologie est réalisée sur les données de l'opération pilote "Delta central du Niger".
Un modèle intégré de l'écosystème du Delta Central du Niger au Mali a été conçu et implémenté. Des algorithmes de vision par ordinateur (traitement d'images) ont été sélectionnées et mis en œuvre.

This document presents the use of the most recent data processing methods (image processing, statistical analysis, symbolic representation) to analyse SIMES data (see Workpackage 1 : "data acquisition"). The validation of our methodology is realised within the pilot operation : "Niger central Delta".
An integrated model of the inner Delta ecosystem has been set up and implemented. Computer Vision algorithms (image processing) has been selected and implemented.

Keyword List:

Integrated ecosystem model, Computer Vision, image processing, statistical analysis, symbolic representation, inner Niger Delta.

A- Introduction

The goal of the deliverable D11 is to show what can be extracted from our specific data (see Workpackage 1 : "Data acquisition") using the most recent data processing techniques (Computer Vision, Statistical Analysis, Symbolic Representation…). We use data from Niger Delta pilot operation whose diversity and quality allow pertinent testing. The work realised within this deliverable include : a state of the art of Data Processing methods, the realisation and implementation of an integrated model of the Delta Niger Ecosystem, the setting up of Computer Vision methods providing relevant information in the context of this model.

B- Organisation

This deliverable is organised as follows :

1. State of the art : Data Processing

1.1 Computer Vision

This section includes a state of the art of Image Processing algorithms relevant within SIMES context. We focus on the extraction of characteristic features from images : image segmentation.

1.2 Statistical Analysis

This section presents a state of the art of the relevant Statistical Techniques for Environmental Information Systems.

1.3 Symbolic Representation

This section deals with a state of the art of the basic Symbolic Representation techniques.

2. Integrated Model for inner Niger Delta in Mali

2.1 Model description

This section describes the integrated model for the inner Niger Delta in Mali that has been realised in the SIMES context.

2.2 User manual

This section is the user manual for the prototype of the integrated model.

3. Extraction of pertinent information from images using Computer Vision techniques

This section describes Computer Vision algorithms and their application to Niger Delta data. It is organised as follows :

3.1 Problematic
3.2 End user requirements
3.3 Computer Vision algorithms classes
3.4 Megawave2 and Targetjunior
3.5 Selection and test of relevant algorithms.

A- Introduction

L'objectif de ce déliverable D11 est de montrer ce qui peut être extrait de nos données spécifiques (voir Worpackage 1 : acquisition de données) avec les outils les plus récents de traitement de données (vision par ordinateur, analyse statistique, représentation symbolique…). Les données utilisées sont les données de l'opération pilote Delta Central du Niger dont la variété et la qualité permettent des tests revêtant une valeur générique. Le travail effectué comprend : un état de l'art des méthodes de traitement de données, la réalisation d'un modèle intégré de l'écosystème du Delta Central du Niger au Mali, la mise en œuvre d'algorithmes de vision par ordinateur fournissant des informations pertinentes pour ce modèle.

B- Organisation

Ce déliverable est organisé comme suit :

1. Etat de l'art : traitement de données

1.1 Vision par ordinateur

Cette partie présente un état de l'art des algorithmes de traitement d'images susceptibles d'être mis en œuvre dans le contexte de SIMES. On se focalisera sur l'extraction d'attributs caractéristiques des images : segmentation d'images.

1.2 Traitements statistiques

Cette section est consacrée à un état de l'art des techniques statistiques pertinentes pour les systèmes d'information pour l'environnement.

1.3 Représentation symbolique

Cette partie est un état de l'art des techniques fondamentales de représentation symbolique

2- Modélisation intégrée du Delta Intérieur du Niger au Mali

2.1 Description du modèle

Ce document décrit le modèle intégré de l'écosystème du Delta Intérieur du Niger au Mali qui a été réalisé dans le cadre de SIMES.

2.2 Manuel d'utilisation

Ce document est un manuel d'utilisation de la maquette du modèle intégré.

3- Extraction d'informations pertinentes des images par des algorithmes de Vision par Ordinateur

Cette partie décrit des algorithmes de vision par ordinateur et leur application aux données du Delta Central du Niger.

1. State of the art : Data Processing

1.1 Computer Vision

This section includes a state of the art of Image Processing algorithms relevant within SIMES context. We focus on the extraction of characteristic features from images : image segmentation.

Projet SIMES

Specification Document SD 2.1.2

State-of-the-art of data processing algorithms

Sébastien Gilles[1], Aurélien Slodzian[2], Pierre Morand[3]

Janvier 1998

[1]Robotics Research Group, Dept of Engineering Science, Oxford University
[2]
[3]

Chapter 1

A survey of image processing techniques

1.1 Introduction

Image processing is the field concerned with all the mathematical operations applicable to a digital image in order to extract or facilitate the extraction of information contained in the image.

In this chapter we review different techniques used by the Computer Vision community, which are relevant to the SIMES project. Emphasis is put on two major issues: (i) Feature-of-interest (especially contour points) detection and (ii) Image segmentation, that is the subdivision of an image in regions.

Advantages and limitations of the different techniques are analysed in order to ease to choice of one or several algorithms during stage DD 2.1.2.

1.2 Sources

The main sources of information used for this document are the books of R. Gonzalez and R. Woods [GW93], O. Monga [Mon93] and O. Faugeras [Fau93]; the bibliography and article servers of Karlsruhe University [1] and South California University [2]; the CVonline server of Edinburgh University [3] and the tutorial site of A. D. Marshall in Edinburgh [4]

1.3 Region segmentation

1.3.1 Split-and-Merge methods

Merging algorithms (see [HP77] for an early reference) use a coarse-to-fine scheme, in which a region is iteratively split until it satisfies a homogeneity

[1] http://liinwww.ira.uka.de/bibliography/Techreports/index.html

[2] http://iris.usc.edu/Vision-Notes/bibliography/contents.html

[3] http://www.dai.ed.ac.uk/CVonline/

[4] http://www.dai.ed.ac.uk/daidb/staff/personal_pages/rbf/CVonline/LOCAL_COPIES/MARSHALL/Vision_lecture_caller.html

criterion. This criterion is generally a statistical parameter. Adjacent regions are merged if they have compatible parameters. Different termination criteria are possible, such as a given number of regions obtained [CJ85], or a predetermined threshold [CBRH83, Cae88] or some statistical rules [SC88, Yak76]. Brice and Fennema [BF77] first partition the image in sets of of points of the same intensity. Then they merge neighbouring regions if a significant fraction of their border have a low contrast.

The main data structure used in those methods is the quadtree. Each parent region is subdivided into 4 sub-images, creating a pyramidal structure. The problem is that final boundaries still reflect the original grid. However a technique by Koepfler [KLM94] proposes to start merging from a single pixel and to use a local energy measure as a merging criteria. In [MYP93], a simulated annealing method is used in order to minimise an energy taking in account different scales. Monga [Mon87] presents a method where the global segmentation quality is optimised through a local optimisation.

1.3.2 Region growing methods

Region growing approach is the opposite of the split and merge approach. Starting from a single pixel (called a *seed*), a region is grown iteratively by adding neighbouring pixels which are compatible with the existing region. This process is repeated until no more pixels can be added.

The simplest criteria for adding a new pixel to an existing region is the difference in grey-level value [Bry79]. Asano and Yokoya [AY81] merge two pixels if the difference of their grey-level values is small compared to the biggest difference between each of the pixels and its neighbours taken in a square neighbourhood. Nagao and Matsuyama [NM83] perform segmentation based on colour. They use the RVB components of a pixel to decide whether to integrate it to an existing region.

The problem with these methods is that a pixel does not carry much information; no global constraint is used.

1.3.3 Snakes and balloons

The use of snakes for extracting image characteristics has been introduced by Kass and *al* [BZ87, KWT88, Ter87a, Ter87b, TWK87]. These methods assume that the boundary of a region or an object is smooth. The deformable model is subject to *internal forces*, which guarantee smoothness and *external forces* which push the model towards the boundary of the object. Recently, Cohen [Coh91] has improved the method by refining the external forces and adding a force which inflates the contour. This method, called *balloons*, ensures that the model is pushed towards the boundary when the external force is not significant enough.

The main problem is to balance the role of the different forces, so that the deformable model locks itself on the real contour. If the inflating force is too strong, the model boundary will be pushed beyond the real contour. If it is too weak, the model boundary will not be attracted to the real contour (if the snake is initially positioned far from the image contours). As a matter of fact, these methods require the user to initialise the contour. The problem of automatical positioning is still an open problem.

1.3.4 Active shape models

Active Shape Models [CCTG91, CT93, CHTH93, CTCG95] are statistical models of the shapes of objects which iteratively deform to fit to an example of the object in a new image. The shapes are constrained by a Point Distribution Model to vary only in ways seen in a training set of labelled examples.

The purpose is therefore to recognise known objects, which seems to be rather unsuitable for segmenting satellite images where shapes tend to be free-form.

1.3.5 Texture segmentation

Texture is a rich source of visual information and has become intensively studied over the last years. This is mainly due to increased popularity of wavelets-based methods for analysing image content. The following methods are particularly adapted to aerial and satellite images.

Texture modelling

Structural texture models consider a texture as the repetition of a 2D pattern (grey-level arrangements, lines, etc...) [Car72, VNP86, BRY93]. Regular textures may be described mathematically by grammars which specify tessellations of the plane [BB82], but texture primitives must be of a regular size and shape, and have a precise spatial arrangement for this model to be accurate.

Bonneh et al. [BRY93] defined a local, generalised symmetry measure which is a continuous quantification of the local spatial relations between edges, and they used the measure to discriminate synthetic texture images. Compared with hierarchical tessellation descriptions, the symmetry measure appears to be useful for texture segmentation because it represents the interactions between primitives quantitatively. The disadvantage of the method is that it only captures the orientational aspects of a texture.

Then textures can be defined statistically by means of its first and second-order moments. First-order statistics turn out to be inefficient for description though very easy to compute. For the second-order statistics, note the use of the co-occurrence matrix. A co-occurrence matrix element p(i,j) is defined at a given separation (d, θ) as the number of pairs of pixels that occur in the image with grey levels i and j, respectively. That is , the location of each component in the matrix gives information about relative grey levels, while the value of the component gives the probability of that relative grey level occurring. The problem is that a co-occurrence can be very heavy to compute for an efficient description. Haralick & al [HKD73] suggest to restrict to five texture descriptors inherited from the co-occurrence matrix. Those five descriptors are described in [Win91].

Also note the use of histograms of absolute grey level differences of pairs of pixels [WDR76, CH80] and generalised co-occurrence matrices [CH80, MFV81].

Transform methods

The most famous of these methods is the Fourier transform which represents the signal in the frequential domain. This technique has been used to segment LANDSAT images [Gra73, HS73]. Once the Fourier transform of an image

has been computed, the angular distribution of the power spectrum reflects the directionality of a texture. Conversely, the radial distribution reflects texture coarseness (cf [GW93]).

Problems arising with this method are well understood and have to do with the fact that an image has an intrinsic finite support; hence its Fourier transform has an infinite support. Also the modulus of the power spectrum does not suffice to describe entirely the signal. An accurate phase information is needed.

Wavelet-based methods

One solution consists in decompose the signal using a well-chosen wavelet basis. The wavelet representation of a signal is both frequentially and spatially localised, which is the main interest of the method.

Before the introduction of wavelets, the Gabor transform was most popular. The Gabor transform uses a basis of Gaussians modulated by a function with finite support [Dau80, You85, Win91, Tur86].

Methods using wavelets give better results (détection, localisation) and are mathematically sound. Recently, the fractal dimension has been suggested as a texture segmentation criterion [CS95, SC92a, XB92, XB93, XB95]. Xie's PhD thesis [Xie94] presents a wavelet-based multiscale analysis and studies thoroughly the concepts of fractal dimension, morphological analysis. The conclusion is that two major texture descriptors are needed for segmentation purposes: local energy and local phase.

Markov field-based methods

Those techniques try to model a textures as a multivariate probability distribution. Given a chosen family of distribution functions, each texture class may be described by selecting an appropriate set of parameters which determine pixel mean, variance and correlation structure [Win91].

It is also possible to use autoregressive models [BL91, MJ74] (a detailed description of these methods can be found in [HS92]), simultaneous autoregressive models (SAR) [PFG78, KB88], Gibbs-Markov models [Win91, DE87, MC91].

1.3.6 Region competition

This method is thought to be best in terms of region segmentation. The original article by Zhu & Yuille [ZYL95] (see also [ZY96] et [ZY95] for a detailed description of the algorithm) proposes a unified theory of segmentation, integrating snakes, balloons and region-growing methods. Once a set of region seeds has been choosen, a set of contours evolves under an MDL[5]-type criterion combining a statistical measure controlling the homogeneity of the different regions, a local information and a regularity constraint (curve smoothness). The algorithm has been implemented in C and C++ at Oxford University. It has been tested on Infrared Linescan images. The code is available to the SIMES project.

[5] Minimum Description Length.

1.4 Edge detection

Contours in an image generally correspond to local extrema of the gradient magnitude or to the zero-crossing of the Laplacian of the intensity surface. The task is made hard because of noise. The problem is therefore to estimate accurately the derivatives of the original signal in the presence of noise.

1.4.1 Early methods

Some of the earliest methods of finding edges in images used small convolution masks to approximate the first derivative of the image brightness function, thus enhancing edges. Roberts [Rob65] used masks of size two pixels by two pixels to find orthogonal derivatives. Masks of odd size, which are easier to implement, have been proposed by Prewitt [Pre70]. Sobel masks [Sob78] are also used for approximating the magnitude of the gradient. These methods were then developed in terms of a computational model of human perception by Marr and Hildreth [MH80, Mar82].

These empirical methods are no longer used as they have severe theoretical and practical limitations.

1.4.2 Mathematical morphology

In [Nob89] Noble uses mathematical morphology to find image structure. Several different morphological operations are described; these are used to enhance edges and find two dimensional features. The "erode-dilate" operator is similar to a first order derivative, and the "open-close" operator is similar to a second order derivative. Good quality edges are found by tracking both sides of each edge and then stitching these "half boundaries" together. Connectivity at junctions is good, although spurious short "tails" sometimes appear at structures such as "T" junctions. The algorithm, including edge tracking, is fairly computationally expensive, and cannot escape the problem of thresholding the feature map.

1.4.3 The Canny edge detector and its variations

The most popular edge detector, introduced by Canny [Can83, Can86], is based on the detection of an ideal step edge corrupted by a Gaussian noise process. Images are first convolved by a finite Gaussian filter. The response is much better than that of a square box (Sobel, Canny, Prewitt). Then, the gradient magnitude map is computed, after which a Non-Maximum suppression step is carried out in order to keep only maxima of the gradient magnitude along the gradient direction. The nice idea about this approach is that two constraints are taken in account to design the edge detector: (1) a model of the kind of edges to be detected, (2) a quantitative definition of the performance of the edge detector.

Deriche [Der87] uses Canny's criteria to derive a different optimal operator. The filter is assumed to have infinite extent this time and is recursively implementable. Another type of filter with infinite extent is derived by [SC86, SC92b].

The limitation of this method and its variations is due to the poor edge model. Most edges in images have complex profiles which differ quite signifi-

cantly from the ideal step edge. The method also tends to perform poorly on T-junctions. The scale of the initial Gaussian filter is user-defined which is an important limitation when information about the scale of objects contained in the image is not known in advance.

1.4.4 Hough transform

The Hough transform is a method for detecting curves of a particular type in an image, which consider global relationships between pixels. The original method is by Hough [Hou62] and deals with line detection, but it has been extented to the case of any parametrisable curve. In [GH88], the authors analyse the sensitivity of the generalised Hough transform in object recognition and address the main problems with this methods, namely the tesselation problem and the sensitivity to occlusions. A good survey is available in [IK88].

1.4.5 Regularisation methods

These methods aims at pre-representing the image before actually extracting the features. Haralick, in [Har84] proposes to approximate the intensity surface by a smooth function (separable Chebychev polynomials). Then the derivatives of the original intensity surface are approximated by the derivatives of the smooth function.

The *weak membrane* approach of Blake and Zisserman [BZ87] involves minimising a global energy function over the image in order to solve for a surface function that fits the image intensity surface. The purpose is to preserve discontinuities.

1.4.6 Anisotropic diffusion

Perona and Malik [PM90] use the heat diffusion equation in order to detect edges. The idea is to smooth the image iteratively and anisotropically. The conduction coefficient is a function of the image gradient: it is high (meaning a strong smoothing effect) when the gradient magnitude is small and vice-versa so that edges are preserved and noise is removed. Over iterations, edges becomes sharper. It is to be noted that features positions remain stable over scale, but the method is quite dependent on local image contrast.

1.4.7 Phase congruency

Using phase congruency for detecting edges in an image appears to be one of the recent major breakthroughs in computer vision. As opposed to the classical spatial filtering techniques (Canny, etc...), the frequential representation of the signal is considered. Postulated by Morrone et *al* [MO87, MB87, MB88], features of interest are defined as points of maximum phase congruency. The importance of phase for human vision was first demonstrated with psycho-visual experiments by Oppenheim and Lim.

Based on this work, Venkatesh and Owens [VO92] showed that points of maximum phase congruency could be calculated equivalently by searching for peaks in the local energy (which is much easier to compute). Later on, Kovesi

[Kov96] proposed a new expression for the phase congruency with an extension to 2D features.

This method gives far better results than the standard Canny edge detector. This is because the approach is essentially multiscale and no predefined scale of analysis is specified. Also no predefined edge model is considered. Different features, such as a roof-like edge, can be picked up by the method. Also, no predefined global threshold is used to extract features (This is a severe limitation for all other types of algorithms).

Implementation details in Matlab can be found in [Kov96].

1.4.8 SUSAN edge detector

SUSAN stands for *Smallest Univalue Segment Assimilating Nucleus*. Consider a circular mask lying on an edge point. Ideally, half of the circular area has the same intensity as the circle's center. This ratio goes down to one fourth for a square angle corner, and even less for sharper angles. Features of interest are thus points in the image where this area (*Univalue Segment Assimilating Nucleus* is minimum.

The SUSAN edge and corner detector [SB94, SB97] uses no image derivatives and is non-linear technique which explains the good tolerance to noise. However, it is conditioned by the choice of a fixed threshold of discriminable contrast.

1.5 Detection of other features

1.5.1 Corners

In [WM92] Wang and Brady develop a curvature-based method. Corner detection is equivalent to detecting maximum curvature points with gradient perpendicular to the edge maximum (two thresholds are thus needed). Corners are detected at different scales as tolerance to noise seems to be a limitating factor.

The SUSAN edge and corner enhancement method [SB94, SB97] has been reviewed in section 1.4.8. Interestingly, this method, which assumes no edge model, can detect junctions, no matter how complex they are.

In [Mor77, Mor79], Moravec developed the idea of using "points of interest", which correspond to locations in the image where a significant change in intensity is detected in every direction. A local autocorrelation is computed and points of interest correspond to a low value. However, the response is anisotropic as, only four directions are used for computing the autocorrelation and the local window is a square. The method is also sensitive to strong edges.

Other methods [Bea78, KR82, DN81, ZH83] involve the computation of the Gaussian curvature along the image intensity surface.

In [HS88], Harris and Stephen use the first-order derivatives to compute the local auto-correlation. It turns out that localisation is good for this method but that localisation is quite poor.

Noble used mathematical morphology in [Nob89] to derive a two-dimensional feature detector. But no results are presented.

1.5.2 Thin nets and crest lines

Crest lines [TG92, TG93, Thi93] are lines where the magnitude maximum curvature (on a surface) is a local maximum in the corresponding principal direction. They have proven of great importance for medical images but have also been applied to road extraction (Thin nets) [MAM97].

These methods require a very good understanding of differential geometry and surface geometry. The problem of these methods is their tolerance to noise as derivatives of the 3rd order are required.

Bibliography

[AY81] T. Asano and N. Yokoya. Image segmentation scheme for low lovel computer vision. In *International Conference on Pattern Recognition*, pages 267–273, 1981.

[BB82] B.H. Ballard and C.M. Brown. *Computer Vision*. Prentice Hall, 1982.

[Bea78] P. R. Beaudet. Rotational invariant image operators. In *International Conference on Pattern Recognition*, pages 579–583, 1978.

[BF77] C.R. Brice and C.L. Fennema. Scene analysis using regions. In *CMetImAly77*, pages 79–100, 1977.

[BL91] Charles Bouman and B. Liu. Multiple resolution segmentation of textured images. *IEEE-PAMI*, 13, 1991.

[Bry79] J. Bryant. On the clustering of multidimensional pictorial data. *Pattern Recognition*, 11:115–125, 1979.

[BRY93] Y. Bonneh, D. Reisfeld, and Y. Yeshurun. Texture discrimination by local generalized symmetry. In *Proceeding of Fourth International Conference on Computer Vision*, pages 461–465, 1993.

[BZ87] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, 1987.

[Cae88] T. M. Caelli. An adaptive computational model for texture segmentation. *IEEE trans. on Systems, Man and Cybernetics*, 18:9–17, 1988.

[Can83] J. Canny. Finding edges and lines in images. In *MIT AI TR*, 1983.

[Can86] J. Canny. A computational approach to edge detection. *IEEE PAMI*, PAMI-8(6):112–130, 1986.

[Car72] L. Carlucci. A formal system for texture languages. *Pattern Recognition*, 4:53–72, 1972.

[CBRH83] T. M. Caelli, H. Brettel, I. Rentschler, and R. Hilz. Discrimination thresholds in the two-dimensional spatial frequency domain. *Vision Research*, 23:129–133, 1983.

[CCTG91] T.F. Cootes, D.H. Cooper, C.J. Taylor, and J. Graham. A trainable method of parametric shape description. *Image Vision Computation*, 10:289–294, 1991.

[CH80] R. W. Conners and C. A. Harlow. A theoretical comparison of texture algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-2:204–222, 1980.

[CHTH93] T.F. Cootes, A. Hill, C.J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. volume 2, pages 33–47. Springer-Verlag, Berlin, New-York, 1993.

[CJ85] J. M. Coggins and A. K. Jain. A spatial filtering approach to texture analysis. *Pattern Recognition Letters*, 3:195–203, 1985.

[Coh91] L.D. Cohen. On active contour models and balloons. *CVGIP*, 53(2):211–218, March 1991.

[CS95] B.B. Chaudhuri and N. Sarkar. Texture segmentation using fractal dimension. *IEEE PAMI*, 17(1):72–77, January 1995.

[CT93] T.F. Cootes and C.J. Taylor. Active shape model search using local grey-level models: A quantitative evaluation. volume 2, pages 639–648. BMVC Press, 1993.

[CTCG95] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active Shape Models - Their training and Application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995.

[Dau80] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Res.*, 20:847–856, 1980.

[DE87] H. Derin and H. Elliot. Modeling and segmentation of noisy and textured images using gibbs random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(1):39–55, 1987.

[Der87] R. Deriche. Using canny's criteria to derive a recursively implemented optimal edge detector. *IJCV*, 1(2):167–187, 1987.

[DN81] L. Dreschler and H.-H. Nagel. Volumetric model and 3d trajectory of a moving car derived from monocular tv-frame sequence of a street scene. In *Computer Vision, Graphics and Image Processing*, volume 20, pages 199–228, 1981.

[Fau93] O.D. Faugeras. Three-dimensional computer vision. *MIT Press*, page 65, 1993.

[GH88] W. Eric L. Grimson and David Huttenlocher. On the Sensitivity of the Hough Transform for Object Recognition. Aim-1044, Artificial Intelligence Laboratory, Massachusetts Institute of Technology (MIT), Cambridge, Massachusetts, May 1988.

[Gra73] N. Gramenopoulos. Terrain type recognition using ests-1 mss images. In *Record of the Symposium on Significant Results obtained from the Earth Resources Technology Satellite, NASA SP-327*, 1973.

[GW93] R.C. Gonzalez and R.E. Woods. Digital image processing. In *Addison-Wesley*, 1993.

[Har84] Robert Haralick. Digital step edges from zero crossing of second directional derivatives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1):58–68, January 1984.

[HKD73] Robert M. Haralick, K.Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE SMC*, SMC-3(6):610–621, 1973.

[Hou62] P.V.C. Hough. Methods and means for recognising complex patterns. Technical report, US patent 3,609,654, 1962.

[HP77] S.L. Horowitz and T. Pavlidis. Picture segmentation by a directed split and merge procedure. In *CMetImAly77*, pages 101–11, 1977.

[HS73] R. J. Horning and A. J. Smith. Application of fourier analysis in multispectral/spatial recognition. In *presented at the Management and Utilization of Remote Sensing Data ASP Symposium*, 1973.

[HS88] C. Harris and M.J. Stephens. A Combined Corner and Edge Detector. In *Alvey88*, pages 147–152, 1988.

[HS92] Robert M. Haralick and Linda Shapiro. *Computer and Robot Vision*. Addison-Wesley Publishing Company, 1992.

[IK88] J. Illingworth and J.V. Kittler. A survey of the hough transform. *CVGIP*, 44(1):87–116, October 1988.

[KB88] A. Khotanzad and A. Bouarfa. A parallel, non-parametric, non-iterative clustering algorithm with application to image segmentation. In *Proceedings, 22nd Asilomar Conference on Signals, Systems and Computers*, pages 305–309, 1988.

[KLM94] G. Koepfler, C . Lopez, and J.M. Morel. A multiscale algorithm for image segmentation by variational method. *SIAM J. Numerical Analysis*, 31(1):282–299, February 1994.

[Kov96] P. Kovesi. *Invariant measures of image features from phase information.* PhD thesis, University of Western Australia, http://www.cs.uwa.edu.au/ pk, 1996.

[KR82] L. Kitchen and A. Rosenfeld. Gray-level corner detection. *Pattern Recognition*, 1:95–102, 1982.

[KWT88] M. Kass, A.P. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, January 1988.

[MAM97] O. Monga, N. Armande, and P. Montesinos. Thin nets and crest lines: application to satellite data and medical images. *Computer Vision and Image Understanding*, 67(3):285–295, sept. 1997.

[Mar82] D.C. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W. H. Freeman, 1982.

[MB87] M. Morrone and D. Burr. Feature detection from local energy. *PRL*, 6:303–313, 1987.

[MB88] M. Morrone and D. Burr. Feature detection in human vision: A phase dependent energy model. *RoyalP*, B-235:221–245, 1988.

[MC91] B. S. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random field models. *PAMI*, 13(5):478–482, 1991.

[MFV81] J. W. Modestino, R. W. Fries, and A. L. Vickers. Texture discrimination based upon an assumed stochastic texture model. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-3:557–580, 1981.

[MH80] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London*, 207:187–217, 1980.

[MJ74] B. H. McCormick and S. N. Jayaramamurthy. Time series model for texture synthesis. *International Journal of Computer and Information Sciences*, 3:1411–1420, 1974.

[MO87] M. Morrone and R. A. Owens. Feature detection from local energy. *Pattern Recognition*, 6:303–313, 1987.

[Mon87] O. Monga. An optimal region growing algorithm for image segmentation. *PRAI*, 1(4):351–375, December 1987.

[Mon93] O. Monga. *Vision par ordinateur: outils fondamentaux*. Hermes, 1993.

[Mor77] H.P. Moravec. Towards automatic visual obstacle avoidance. In *International Joint Conference on Artificial Intelligence*, 1977.

[Mor79] H.P. Moravec. Visual mapping by a robot rover. In *International Joint Conference on Artificial Intelligence*, 1979.

[MYP93] R. Muzzolini, Y.H. Yang, and R. Pierson. Multiresolution texture segmentation with application to diagnostic ultrasound images. *IEEE Trans. on Medical Imaging*, 12(1):108–121, March 1993.

[NM83] M. Nagao and M. Matsuyama. *Structural Image Analysis*. Morton Nadler, 1983.

[Nob89] J. Alison Noble. *Description of Image Surfaces*. PhD thesis, University of Oxford, 1989.

[PFG78] W. K. Pratt, O. D. Faugeras, and A. Gagalowicz. Visual discrimination of stochastic texture field. *IEEE Trans. on Systems, Man and Cybernetics*, SMC-8:796–804, 1978.

[PM90] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *PAMI*, 12(7):629–639, July 1990.

[Pre70] J.M.S. Prewitt. Object enhancement and extraction. In Lipkin, B.S. and Rosenfeld, A., editor, *Picture Processing and Psychopictorics*, pages 75–149. New York: Academic, 1970.

[Rob65] L.G. Roberts. Machine perception of three-dimensional solids. In Tippett, J. and Berkowitz, D. and Clapp, L. and Koester, C. and Vanderburgh, A., editor, *Optical and Electrooptical Information processing*, pages 159–197. MIT Press, 1965.

[SB94] Stephen M. Smith and J.M. Brady. Susan - a new approach to low level image processing. Technical Report TR95SMS1, Defense Research Agency, Farnborough, Hampshire, UK, 1994.

[SB97] S.M. Smith and J.M Brady. Susan - a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, May 1997.

[SC86] J. Shen and S. Castan. An optimal linear operator for edge detection. In *CVPR86*, pages 109–114, 1986.

[SC88] J. F. Silverman and D. B. Cooper. Bayesian clustering for unsupervised estimation of surface and texture models. *IEEE-PAMI*, 10:482–495, 1988.

[SC92a] N. Sarkar and B. B. Chaudhuri. An efficient approach to estimate fractal dimension of textural images. *Pattern Recognition*, 25(9):1035–1041, 1992.

[SC92b] J. Shen and S. Castan. An optimal linear operator for step edge detection. *GMIP*, 54(1):112–133, 1992.

[Sob78] I. Sobel. Neighbourhood Coding of Binary Images for Fast Contour Following and General Array Binary Processing. *Computer Vision, Graphics and Image Processing*, 8:127–135, 1978.

[Ter87a] D. Terzopoulos. On matching deformable models to images. *Topical meeting on machine vision, Technical Digest Series*, 12:160–163, 1987. Optical Society of America.

[Ter87b] D. Terzopoulos. On matching deformable models to images: direct and iterative solutions. *Topical meeting on machine vision, Technical Digest Series*, 12:164–167, 1987. Optical Society of America.

[TG92] J.P. Thirion and A. Gourdon. The 3d marching lines algorithm and its application to crest lines extraction. Technical report 1672, INRIA, 1992.

[TG93] J.P. Thirion and A. Gourdon. The 3d marching lines algorithm: New results and proofs. Technical report 1881, INRIA, 1993.

[Thi93] J.P. Thirion. The extremal mesh and understanding of 3d surfaces. Research report 2149, INRIA, December 1993.

[Tur86] M. R. Turner. Texture discrimination by gabor functions. *Biol. Cybernet.*, 55:71–82, 1986.

[TWK87] D. Terzopoulos, A.P. Witkin, and M. Kass. Symmetry-seeking models and 3d object reconstruction. In *International Conference in Computer Vision*, pages 269–276, 1987.

[VNP86] F. M. Vilnrotter, R. Nevatia, and K.E. Price. Structural analysis of natural textures. *IEEE-PAMI*, PAMI-8, 1986.

[VO92] S. Venkatesh and R. A. Owens. An energy feature detection scheme. In *Proceedings of 2nd International Conference on Image Processing*, volume 1, 1992.

[WDR76] J.S. Weszka, C. R. Deya, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Trans. on Systems, Man and Cybernetics*, SMC-6:269–285, 1976.

[Win91] C. Philip Winder. *Markovian Analysis of Texture: Serial and Parallel Paradigms in Low-Level Vision.* PhD thesis, University of Oxford, 1991.

[WM92] W.J. Wang and P. D. McFadden. Time-frequency domain analysis of vibration signals for machinery diagnostics: The present power spectral density. *Technical Report OUEL 1911/92, University of Oxford*, 1992.

[XB92] Zhi-Yan Xie and Michael Brady. Fractal dimension image for texture segmentation. In *Proceedings of 2nd International Conference on Automation, Robotics and Computer Vision*, volume 1, pages CV–4.3.1 to CV–4.3.5, 1992.

[XB93] Zhi-Yan Xie and Michael Brady. Wavelet multi-scale representation and morphological filtering for texture segmentation. In *IEE Colloquium'93: Morphological and Nonlinear Image Processing Techniques*, 1993.

[XB95] Zhi-Yan Xie and Michael Brady. 2d phase-independent local features for texture segmentation. In *ICIP'95 (Inter. Conf. on Image Processing)*, volume 3, page 640 to 643, Washington D.C., Oct 1995.

[Xie94] Z. Xie. *Multi-scale Analysis and Texture Segmentation.* PhD thesis, University of Oxford, 1994.

[Yak76] Y. Yakimovsky. Boundary and object detection in real world images. *Journal of the ACM*, 23:599–618, 1976.

[You85] R. Young. The gaussian derivative theory of spatial vision: analysis of cortical cell receptive field line-weighting profiles. *Technical Report GMR-4920, General Motors Research, Warren, Mich.*, 1985.

[ZH83] O.A. Zuniga and R.M. Haralick. Corner detection using the facet model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 30–37, 1983.

[ZY95] S.C. Zhu and A. Yuille. Region competition and its analysis: A unified theory for image segmentation. Technical report, Harvard Robotics Lab., 1995. No. 95-07.

[ZY96] S.C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *PAMI*, 18(9):884–900, September 1996.

[ZYL95] S.C. Zhu, A. Yuille, and T.S. Lee. Region competition: Unifying snakes, region growing, and bayes/mdl for multi-band image segmentation. In *ICCV95*, pages 416–423, 1995.

1.2 Statistical Analysis

This section presents a state of the art of the relevant Statistical Techniques for Environmental Information Systems.

ETAT DE L'ART EN MODELISATION STATISTIQUE

Les techniques statistiques et leur utilisation

(contribution au délivrable n° 2.1.2 de la convention SIMES -

rédacteur: Pierre Morand)

Abstract: By this text, we skim over of the most useful statistical methods according to the prospect of setting up information systems for environment. The first part reminds the existence of several possible ways to classify statistical methods. The second part describes briefly principles and usage of the main methods. At first, we tackle sampling and estimation techniques and we approach the distribution description of a quantitative or qualitative variable in a sample, then we describe several techniques which aim to measure or to test relationships between variables. In a second place we review exploratory multivariates techniques as clustering and ordination, as well as temporal data series analysis. All over the previous points, we have focused on criteria conditionning the choice of techniques without concealing difficulties in the practice of data processing.

Nombreux et variés, les thèmes et objets susceptibles d'être la cible d'un observatoire de l'environnement ne sont en général pas nouveaux; de ce fait, ils ont déjà été l'objet d'études quantitatives. Les statisticiens, biométriciens et informaticiens qui ont participé à ces études ne s'inscrivaient pas forcément dans une logique de mise en place d'observatoire, mais ils ont compris depuis longtemps que l'adaptation des statistiques aux terrains ouverts et non contrôlés nécessitait une réflexion et des développements particuliers, ce dont témoigne l'existence de nombreux manuels généralistes dédiés (**12**, **13**, **15**, **19**). C'est pourquoi des solutions ou des savoir-faire existent déjà dans la plupart des cas. Le but de ce document est d'en faire un inventaire et de les présenter brièvement, en insistant, le cas échéant, sur l'intérêt spécifique qu'ils peuvent présenter par rapport à une démarche d'observatoire ou de système d'information sur l'environnement. Ce faisant, on cherchera à faire apparaître les avantages, difficultés et limites d'application dans ce même contexte.

Pour exposer un ensemble de techniques statistiques[1], il faut bien procéder par ordre en adoptant une entrée logique, en d'autres termes une catégorisation. Or il est presque impossible de classifier ces techniques de façon univoque et satisfaisante pour tout le monde. C'est pourquoi, après avoir exposé un certain nombre de critères et repères généraux, nous nous en tiendrons à une revue très conventionnelle ordonnée *par famille d'usage*.

1 - Divers critères de classifications des techniques statistiques.

1.1. Critère contextuel: le domaine d'application.

En général, la contrainte la plus forte, pour étudier ou suivre le déroulement d'un processus, est celle qui porte sur la méthode d'observation, et quelquefois même sur la possibilité, pas toujours évidente, de récolter de l'information à un coût non prohibitif. La latitude de choix est généralement faible sur cette étape. Or la méthode de recueil de l'information, une fois adoptée, va conditionner dans une large mesure le format d'information produit. Nous qualifierons ici de *domaine d'application* un certain type de méthode d'observation associé au format de données qu'il engendre habituellement. On verra que ces domaines d'application sont très spécifiques, tant par les hypothèses qui conditionnent leur valeur d'interprétation que par le format des données lui-même. Par conséquent, la plupart des outils de traitement et d'extraction de résumés statistiques vont être en pratique dédiés à un ou deux

[1] A l'expression un peu pompeuse de 'modèles statistiques' nous avons préféré celle, plus humble, de 'techniques statistiques', étant entendu que ces techniques sont presque toujours appuyées sur des modèles. Cependant, la définition d'une technique étant liée à un certaine perspective opérationnelle, donc à un certain contexte, il est clair que le nombre de techniques que l'on peut citer est supérieur à celui des modèles auxquels elles font appel.

domaines d'application. De ce fait, il serait tout à fait possible de proposer une première *classification très pragmatique* des techniques statistiques en adoptant comme entrée le domaine d'application. Nous proposons la liste suivante de domaines d'application des techniques statistiques:

- Les expériences planifiées (*i.e.* contrôlées), courante en agronomie et en essai clinique (I)
- Les registres et les recensements administratifs (II)
- Les enquêtes par échantillonnage sur les opinions ou sur les comportements (III)
- Les inventaires ou dénombrements d'espèces biologiques par stations, placettes (IV)
- Les séries de dosages ou d'enregistrement sur des points de mesures fixes (V)
- Le suivi au cours du temps du comportement d'individus identifiés (études "longitudinales" en démographie, épidémiologie…) (VI)
- Les informations "en couverture aréale continue" (images satellitales, photos aériennes) (VII)

Par rapport au contexte du projet SIMES, à savoir le développement d'outils pour les observatoires, la première forme d'acquisition d'information ne présente qu'une très faible pertinence. Nous ne la traiterons donc pas dans ce document. Enfin, le dernier domaine fait l'objet d'un autre document SIMES et nous laisserons également de côté les outils statistiques qui peuvent lui être associées. Ce sont donc les techniques statistiques applicables aux domaines ci-dessus numérotés 2 à 6 que nous évoquerons ici.

1.2. <u>Critère fonctionnel: place dans le processus de recherche.</u>

Les techniques statistiques peuvent intervenir à différentes étapes du processus de recherche, et leur intervention peut répondre à des objectifs ou à des formes de question variées. Ce constat nous amène à définir une classification selon un critère fonctionnel. Par rapport à la place dans le processus de recherche et dans le cycle de l'information, nous distinguerons donc:

- les techniques qui interviennent comme aide à la planification de l'acquisition des données (P)
- les techniques qui sont utilisées pour traiter (analyser, synthétiser…) les données (T)

Dans ce dernier cas, qui est de loin les plus fréquent, le rôle de la technique statistique peut être:

- soit d'aider à explorer des masses de données complexes, à y rechercher et à y mettre en évidence des tendances (T1).
- soit d'extrapoler, d'estimer ou de prédire, avec un certain niveau de précision, une valeur ou un indice (T2)
- soit de tester la validité d'une hypothèse, pour pouvoir l'accepter ou la rejeter avec un certain niveau d'assurance (T3)

Il faut noter d'emblée que la poursuite de l'une ou l'autre de ces trois classes d'objectifs doit être très clairement et explicitement annoncée en amont de tout travail statistique. En effet, choisir un objectif sur le traitement d'un lot de données peut impliquer de renoncer à un autre. Le cas exemplaire est celui de l'incompatibilité entre les démarches exploratoires (T1) et les démarches "confirmatoires" (T3), car on ne peut tester et confirmer une hypothèse sur le lot de données qui a permis de la faire naître. Ce point délicat de la démarche statistique nécessite d'autant plus de prudence que certaines techniques statistiques peuvent servir, à la demande, différents objectifs - par exemple, on verra plus loin que la *régression linéaire* est une technique utilisée tantôt pour répondre à des objectifs de type T1, tantôt pour répondre à des objectifs de type T2 ou même T3.

1.3. Critère de qualité de constitution: relation avec la théorie statistique.

Répondant aux demandes et aux besoins très variés des chercheurs de toutes les disciplines, les statisticiens développement continuellement de nouvelles techniques. Certaines d'entre elles s'appuient directement sur la théorie statistique et sur ses expressions en forme de lois. D'autres techniques sont moins orthodoxes mais parviennent à s'imposer par leurs qualités de simplicité ou d'efficacité empirique. Nous pouvons utiliser cette relation plus ou moins étroite et solide avec la théorie statistique comme une troisième entrée de classification des techniques. Nous distinguerons donc, du centre vers la périphérie:

- A: des techniques qui *mettent en œuvre* ou participent à l'avancement de la théorie elle-même. C'est souvent le cas avec la modélisation des distributions (populations, processus temporels et spatiaux).

- B: des techniques qui sont *utilisatrices* des résultats et de la théorie statistique. Leur utilisation nécessite généralement qu'un certain nombre de conditions, portant sur la taille des échantillons et/ou sur la forme de distribution des variables traitées, soient réunies. On citera les techniques de l'échantillonnage et de l'estimation, les techniques de modélisation des relations entre variables (modèle linéaire avec critère), les techniques permettant de tester des hypothèses. Toutes ces techniques pourraient être ventilées plus finement en un certain nombre de familles en fonction des résultats ou des hypothèses de la théorie statistique auxquels elles se réfèrent.

- C: des techniques qui servent à *contourner* les difficultés d'application de la théorie statistique. Elles tentent de se substituer aux méthodes de classe (B) lorsque les conditions d'applicabilité de celles-ci ne sont pas réunies. Elles font appel à des techniques de randomisation "Monte-Carlo" (reproduction et démultiplication de l'effet du hasard). Parmi les plus connues, on citera les techniques de *bootstrap* et de *jacknife* (**11**).

- D: des techniques qui, bien que mettant en jeu des calculs numériques, ne font pas référence à la théorie statistique. Il s'agit alors de simples techniques de traitement de données mais pas de techniques *statistiques* au sens fort du terme, dans la mesure où la notion de "quantité statistique dotée d'une distribution" en est absente. Par exemple, on qualifiera ainsi certains indices utilisés pour comparer des formes ou des profils à partir de calculs d'inspiration géométrique, certaines analyses multivariées ou bien certaines techniques d'identification ou de paramétrage de modèles par essais, erreurs et pénalités. Ces techniques ne sont pas d'un usage rare.

Il faut noter que la place occupée à un moment donné par une technique dans ce système de classification concentrique n'est pas définitive. Par exemple, une technique très empirique, reléguée au début de son histoire dans la zone périphérique "D", peut, du fait de son succès, intéresser des statisticiens, lesquels vont s'attacher à en décrire les ressorts (en termes probabilistes) et vont ensuite décrire les distributions théoriques (attendues sous différentes hypothèses) des quantités calculées par la technique (que ce soit en *output* ou lors des étapes intermédiaires). A partir de ce moment, la dite technique rejoint l'orthodoxie statistique. C'est ainsi que les *modèles en réseaux de neurones* (**16**) classés à leurs débuts comme de géniaux bricolages d'informaticiens, sont venus aujourd'hui rejoindre et enrichir la théorie du modèle linéaire (cf. § 2.2.3 ci-après).

2 - Revue des classes de techniques et modèles statistiques, classés de façon conventionnelle.

2.1. L'échantillonnage et l'estimation.

Hormis le cas particulier des recensements exhaustifs (qui sont le plus souvent d'origine administrative), les études scientifiques de terrain impliquent en général une procédure d'échantillonnage, c'est-à-dire une procédure par laquelle on va choisir une *fraction d'éléments* d'un système (ou univers), fraction sur laquelle va se concentrer l'effort d'observation et de description (**26**). Les informations extraites de l'échantillon vont ensuite être utilisées pour donner une appréciation sur l'ensemble du système (ou univers) étudié. Ce double mouvement de resserrement (de l'effort de description) puis d'extension (des résultats et conclusions) soulève de nombreuses questions qui ont conduit à la construction progressive d'une théorie à laquelle se rattache tout un ensemble de techniques. Ces techniques répondent à deux questions pratiques:

- comment, pour un budget de recherche donné, choisir au mieux la fraction d'éléments qui subira l'effort de description ? C'est la question de *l'optimisation de l'échantillonnage*.

- comment, à partir des caractéristiques connues d'un échantillon, parvenir à des résultats qualifiant l'ensemble du système ? C'est la question de *l'estimation inférentielle*, souvent appelée extrapolation.

Ces deux questions sont traitées conjointement par la théorie de l'échantillonnage (**5**, **10**).

2.1.1. Constitution d'échantillon et estimation dans le cadre de la théorie de l'échantillonnage aléatoire.

Pour que les résultats obtenus sur un échantillon soient généralisables à la population, l'échantillon doit être représentatif de cette dernière, c'est-à-dire qu'il doit refléter fidèlement sa composition et sa complexité. La meilleure façon d'assurer la représentativité de l'échantillon est d'employer, pour sélectionner les unités ou éléments échantillonnés, un processus de sélection de type *aléatoire*. Pour cela, le plan de collecte doit être tel que *chaque élément de l'échantillon a une probabilité connue et non nulle d'appartenir à l'échantillon*.

Le plus connu des plans d'échantillonnage aléatoire est le plan de type "e.a.s." (échantillonnage aléatoire simple) qui consiste à prélever au hasard et de façon indépendante n éléments d'une population de N éléments. Chaque élément de la population possède la même probabilité (p = n/N) d'appartenir à l'échantillon. Une fois constitué, l'échantillon sera décrit par le calcul de paramètres comme la moyenne, la quantité (somme), la proportion ou l'écart-type (cf. § 2.2.1), pour une variable donnée. Puis on utilise des formules (estimateurs) pour inférer ces résultats à la population toute entière.

Un autre type de plan, dit échantillonnage systématique, consiste établir une liste des éléments de la population, puis à tirer au hasard un i^{eme} élément situé entre le premier et le p^{ieme} élément de la population, puis à prélever systématiquement les $(i + mp)^{ieme}$ éléments de la liste, où m est un entier prenant les valeurs de 1 à n et où p est le pas d'échantillonnage. Ce plan est particulièrement adapté pour sélectionner des éléments régulièrement disposés sur un axe temporel ou spatial.

Le plan à plusieurs degrés repose sur un système ramifié et hiérarchisé d'unités. Les unités supérieures (les plus grosses et les moins nombreuses) sont appelés unités primaires. Un premier échantillonnage aléatoire a lieu à leur niveau. A l'intérieur de chaque unité primaire, on reconnaît des éléments plus fins (unités secondaires) dont on va dresser la liste et sur lequel va porter un autre processus d'échantillonnage. Il peut exister ainsi jusqu'à trois ou quatre niveaux. Ce genre de plan est utile lorsque l'on ne peut pas, au départ, dresser directement la liste des éléments les plus fins. Il a aussi l'avantage de pouvoir coller de près à la structure naturelle des systèmes sociaux ou écologiques (ex.: les villages contiennent des quartiers qui contiennent des ménages qui possèdent des parcelles…). Ces plans sont très courants et peuvent s'avérer particulièrement utiles dans le cadre d'un observatoire, mais ils entraînent tout de même quelques difficultés dans la manipulation et l'analyse de l'information collectée (cf. § 2.3.3).

Le plan stratifié consiste à partitionner préalablement l'univers étudié, avant d'appliquer dans chaque partie (*strate*) l'un des plans précédents, sans forcément être obligé de choisir le même type de plan pour toutes les strates. Les formules d'estimateurs correspondantes sont toujours lourdes, et nombre d'analyses statistiques deviennent difficiles ou même biaisées lorsqu'on veut les appliquer sur des données acquises à travers un tel plan. Mais le recours à ce genre de plan est indispensable si l'on a affaire à un univers très hétérogène.

Dans tous les cas, la formulation des estimateurs est indissociable de la configuration du plan d'échantillonnage.

En conclusion de ce §, on dira que les techniques de planification d'échantillon aléatoire (et les estimateurs associés) sont particulièrement efficaces dans le cadre d'une "approche objectif", où l'on pose *au départ* un nombre *restreint* de résultats bien précis à produire. Par exemple, ces techniques montreront toute leur puissance dans le cadre d'une étude visant à évaluer le pourcentage d'arbres malades dans une forêt ou bien à estimer la quantité totale d'un certain produit polluant dans les eaux d'un lac. Par contre, ces techniques sont d'un intérêt moindre pour une étude exploratoire systémique et/ou pluridisciplinaire, car le nombre des questions et la multiplicité des variables (mesures, comptages, questions…) empêchent toute démarche d'optimisation formelle de la collecte. Mieux vaut alors s'en tenir à quelques principes de bon sens pour la définition du plan, qui a intérêt à rester simple. Toutefois, pour peu que ce plan de collecte soit explicite et respectueux de quelques règles de base (visant notamment à préserver le caractère aléatoire et non biaisé de la constitution de l'échantillon), il sera toujours possible de puiser dans l'arsenal de la théorie de l'e.a. les formules adaptées pour calculer les estimations et leurs précisions. Mais il ne faudra pas s'étonner de constater que les intervalles de confiance obtenus sont bien souvent très larges, suscitant critiques et incompréhensions de la part d'un public non averti.

2.1.2. Les techniques d'échantillonnage par quota.

Nous avons vu ci-dessus que la théorie de l'échantillonnage aléatoire ne trouve pas toujours à s'employer dans toute sa puissance, dès lors que les variables collectées sont multiples. Mais il est d'autres entraves, plus graves, à son utilisation. Les plus importantes sont liées à sa mise en œuvre. En effet, le principe du tirage aléatoire suppose qu'à tout niveau du plan, on dispose d'une liste exhaustive (dite liste ou base de sondage) des éléments du niveau immédiatement inférieur et que cette liste doit contenir les informations relatives au(x)critère(s) de stratification (par exemple, s'il y a une stratification sur la variable sexe, il faut la liste des individus avec la mention de leur sexe). Or, des listes aussi bien documentées et à jour ne sont pas forcément disponibles. En supposant même que cette liste soit acquise, les difficultés ne sont pas terminées, car il faut maintenant sélectionner les éléments constitutifs de l'échantillon par une procédure de tirage et aller ensuite observer ou visiter les éléments sélectionnés, et ceux-ci seulement. Or, dans bien des cas, il est difficile et fort coûteux de programmer une rencontre avec un individu précis (surtout s'il s'agit d'un être humain !). Pour contourner tous ces problèmes d'application de l'échantillonnage aléatoire, les statisticiens des instituts de sondage préfèrent adopter une démarche moins orthodoxe mais dont l'efficacité est empiriquement avérée: *l'échantillonnage raisonné*, dit encore "*par quota*" (**10**). Il s'agit de favoriser avant tout le processus des rencontres successives entre l'enquêteur et les individus à enquêter, et de vérifier au fur et à mesure de l'avancement de ce processus que l'accumulation des rencontres déjà effectuées ne s'écarte pas trop de *quotas* de composition fixés au départ pour la composition de l'échantillon (ces quotas sont définis pour assurer au sein de l'échantillon une certaine représentation des modalités de différentes variables: sexe, âge, profession etc..). Si un écart apparaît, l'enquêteur cherche à orienter davantage les rencontres suivantes sur des individus présentant les caractéristiques insuffisamment représentées dans le début du parcours d'enquête. Seules les dernières rencontres sont parfois difficiles

à réaliser, puisqu'elles doivent correspondre à des modalités précises. Mais il existe aussi des techniques de redressement d'échantillon (**10**) qui permettent de corriger, dans une certaine mesure, d'éventuelles écarts entre les quotas fixés au départ et la composition de l'échantillon effectif issu du travail d'enquête.

Les faits montrent que les sondages par quota sont particulièrement efficaces pour évaluer des paramètres tels que des moyennes ou des proportions dans une population. Toutefois, l'inconvénient du sondage par quota est qu'il ne permet pas le calcul d'estimateurs statistiques formels assortis de niveau de précision, dans la mesure où l'intervention du hasard dans la constitution de l'échantillon ne se fait pas de façon "pure". Pour cette raison, les chercheurs méprisent bien souvent (et sans doute à tort) cette technique, alors que les statisticiens des instituts de sondage, davantage soumis à des contraintes d'efficacité, en ont fait depuis longtemps leur arme favorite en matière de suivis des opinions politiques et des modes de consommation (études de *marketing*).

2.1.3. Les problèmes de l'échantillonnage et de l'inférence relèvent-ils toujours d'une approche statistique en termes de population/échantillon ?

Cette question peut sembler surprenante, mais elle surgit rapidement lorsque l'on se trouve confronté à des travaux de terrain. En effet, les notions de population et d'échantillon, telles que définies dans les § précédents, conviennent mal aux études sur l'environnement physico-chimique. Par exemple, il est difficile de considérer les vingt cercles de bouches de pluviomètres disposés sur la surface d'un bassin versant comme un échantillon de la population des cercles de même taille contenus dans le territoire de ce bassin. Et que dire alors des vingt thermomètres que l'on aura placé à côté des pluviomètres ? Pourtant, le besoin de réaliser des inférences (par exemple pour calculer la quantité totale de pluie qui tombe dans le bassin, ou bien la température moyenne qui y règne) est tout aussi fort dans ce domaine qu'en écologie, en démographie ou en socio-économie. Mais l'inférence passe ici nécessairement par l'utilisation, souvent implicite, de modèles de continuité postulant que des points voisins ont des valeurs proches. Certaines techniques comme le *krigeage* que nous verrons plus loin (§ 2.5.1), rendent ces hypothèses explicites et permettent d'aboutir à des estimations inférentielles argumentées.

2.2. L'étude fine des distributions de variables observées et des relations entre ces distributions.

L'étude des distributions (ainsi que de multiples questions connexes rattachées à ce thème) a toujours été le sujet principal d'attention des statisticiens, et il ne faut donc pas s'étonner de trouver ici le plus grand nombre de modèles et de techniques statistiques. On rappellera qu'une distribution est constituée par la compilation et la représentation, par un moyen quelconque, de l'ensemble des valeurs observées, pour une variable, sur les individus d'une population donnée ou sur un échantillon d'individus de cette population (ce dernier cas étant évidemment le plus fréquent). On parlera par exemple de la distribution de la variable "poids" dans la population d'un pays, ou bien dans un échantillon de personnes de cette population.

De façon historique, la modélisation des distributions est au coeur de la théorie statistique, avec les travaux fondateurs de Bernouilli et de Gauss. Nous ne décrirons pas ici le contenu de ce socle théorique, solidement ancré dans la théorie des probabilités, qui sert de base ou du moins de repère au foisonnement de techniques dont nous allons essayées de brosser un panorama dans les paragraphes suivants.

2.2.1. Techniques de caractérisation d'une distribution.

Ces techniques sont enseignées dans les ouvrages généraux (**12**, **26**) comme statistiques descriptives "de base" à l'usage d'une large gamme d'utilisateurs. Leur parfaite maîtrise reste pourtant nécessaire à tout professionnel de la statistique, car c'est par ces techniques que passe la communication de bons nombres de résultats, que ce soit avec d'autres chercheurs ou avec le public. La description d'une distribution s'appuie sur deux formes de langage: l'expression de paramètres synthétiques et la représentation graphique. Pour aller plus loin sur ce point, il faut distinguer deux cas:

a) le cas des variables qualitatives (dites encore nominales)

Une variable qualitative est un caractère qui peut se présenter sous forme d'une liste finie et brève de valeurs (dites "modalités") mutuellement exclusives (ex.: la couleur des yeux peut prendre l'une des cinq valeur suivantes: bleu ou brun ou noir ou vert ou gris). Le diagramme en bâton (ou *chart*) et le diagramme en camembert, presque équivalent, sont les représentations graphiques les plus adaptées: à chaque modalité que peut prendre la variable, on associera l'effectif ou le pourcentage de cas observés. Il n'est pas très utile de produire des paramètres synthétiques sur ce genre de distribution, si ce n'est le pourcentage d'occurrence de cas dans la modalité la plus représenté, la richesse totale (nombre de modalités présentant au moins un cas) ou bien encore la diversité, qui est d'autant plus forte que la richesse est élevée et que les effectifs sont assez équitablement répartis entre les modalités.

b) le cas des distributions de variables quantitatives

Les variables quantitatives présentent un domaine de variation continu (mesures…) ou quasi continu (comptages..), avec une possibilité d'établir un ordre (de type "supérieur à" ou "inférieur à") entre les valeurs observées. La description de leur distribution a donné lieu au développement d'un langage très riche. La représentation de base est l'histogramme qui est un diagramme en bâton dans lequel les modalités, représentées par les classes ou intervalles successifs de valeurs possibles, ne sont pas permutables. On peut lui associer la fonction de répartition, ou histogramme cumulée. De nombreux paramètres visent à décrire l'histogramme et son allure:

- certains paramètres ont place au cœur de la théorie statistique, ce sont la moyenne et les moments d'ordre 2, 3 ou plus (somme des écarts à la moyenne affectés d'un exposant respectivement carré, cube etc…). Le moment d'ordre 2 est plus connu sous le nom de *variance*.

- d'autres paramètres ont un intérêt pratique évident: le mode (intervalle de valeurs pour lequel on observe le plus grand nombre de cas) et l'étendue (écart entre la valeur min. et la valeur max. prises par la variable), c'est-à-dire la hauteur et la largeur de l'histogramme.

- enfin, certains paramètres sont utilisés parce que considérées comme robustes vis à vis de la présence de données aberrantes ou erronées. Ce sont notamment les percentiles (médiane, quartile, décile) (12). La médiane est la valeur de la variable qui laisse 50% des observations à sa gauche et 50% des observations à sa droite dans l'histogramme. Le premier quartile laisse 25 % de valeurs à sa gauche et 75 % à sa droite.

- en examinant l'histogramme et certains paramètres synthétiques, on pourra qualifier l'allure de la distribution. L'allure attendue (et souhaitée) est une allure "en cloche" dite en forme de *loi normale* (ou loi de Gauss). Si la distribution est parfaitement normale, elle est entièrement décrite par la seule connaissance de sa moyenne et de sa variance. Ce n'est bien sûr jamais tout à fait le cas. Les moments d'ordre 3 et 4 servent à construire des indices, comme le *kurtosis*, qui mesurent le degré

d'irrégularité de la distribution par rapport à une forme normale (excès d'aplatissement, asymétrie plus ou moins marquée etc…).

On peut admettre que tout ce qui a été dit ci-dessus concerne les distributions observées, correspondant en général aux données d'un échantillon (ou, plus rarement, d'un recensement). Si l'on veut maintenant induire, à partir de ces informations, une connaissance sur la distribution "mère" (celle de la population), il faut faire appel à la théorie de l'échantillonnage et de l'estimation (§ 2.1.1.). Celle-ci permet d'estimer, à partir des paramètres de la distribution d'une variable observée sur un échantillon, les paramètres de la distribution de la variable dans la population dont est issu l'échantillon.

2.2.2. Comparaisons de distributions d'une même variable dans plusieurs échantillons (**26**).

Le problème est le suivant: deux échantillons, caractérisées par des valeurs observées pour une même variable, sont-ils suffisamment différents pour pouvoir considérer qu'ils proviennent de populations différentes: c'est le test de la *différence des moyennes* (test t de Student), où l'hypothèse nulle à rejeter est celle de l'absence de différence des moyennes entre les deux populations. On peut étendre le problème à plus de deux échantillons et l'on obtient ainsi *l'analyse de variance* à un facteur, dite "test d'homogénéité des moyennes". On peut aussi tester l'homogénéité des variances de deux échantillons. Nous voyons donc que la comparaison de distributions est un domaine qui se traduit généralement par une approche en termes d'analyse confirmatoire (test d'hypothèse).

2.2.3. Interdépendances entre les distributions de plusieurs variables observées sur un même échantillon d'éléments. Introduction au modèle linéaire.

C'est le domaine le plus riche des statistiques. De multiples techniques sont disponibles pour pouvoir aborder tous les cas de figure possibles ou choisies:

- Toutes les variables impliquées dans l'analyse sont-elles quantitatives ? ou bien qualitatives ? ou bien de natures différentes ?

- S'agit-il de deux variables seulement ou bien d'un nombre un peu plus grand (3, 4, 5…) ?

- Veut-on tester l'existence de relations ? mesurer (quantifier) leur intensité ? ou bien encore prédire une variable en *output* à partir d'une ou plusieurs autres connues en *input* ?

Prenons le cas où deux variables quantitatives sont en cause, dont les valeurs sont connues sur un échantillon aléatoire. Le coefficient de *corrélation* de Pearson permet à la fois de tester l'existence d'une relation entre les variations de ces deux variables (rejet de l'hypothèse d'indépendance) et d'évaluer le degré d'intensité et le sens de cette relation. Si l'on veut maintenant faire jouer à l'une des deux variables le rôle de "variable X explicative" et à l'autre le rôle de "variable Y à prédire", alors le problème devient asymétrique et l'on dispose d'une technique, dite *régression linéaire* (**9**, **27**), qui permet d'ajuster une fonction linéaire $Y=aX + b$, ou a et b sont des paramètres estimés de façon à minimiser la somme des écarts quadratiques entre les valeurs y prédites et les valeurs y observées. De façon analogue, si le nombre de variables est égale à 3, 4 , 5 … z et que l'on veut prédire l'une d'entre à partir des autres, on utilise la *régression linéaire multiple* (**9**, **27**) qui, dans la fonction $Y=a_1X_1+ a_2X_2 + \ldots a_{z-1} X_{z-1} + b$, permet d'estimer le vecteur de paramètres $(a_1, a_2\ldots, a_{z-1}, b)$ minimisant la somme des *écarts quadratiques* entre les valeurs y prédites et les valeurs y observées.

Lorsque l'on a affaire à deux variables qualitatives observées sur les éléments d'un échantillon et que l'on veut tester l'existence d'une relation (on parlera alors d'*association*) entre les distributions de ces deux variables, alors on établit le tableau croisé des effectifs (ou *tableau de contingence*), sur lequel on applique le test du _^{2} (Chi-carré), ou bien le test exact de Fisher (**19**, **26**) si le tableau de contingence est de format 2_2 et que les effectifs des cellules sont faibles. Pour mesurer l'intensité de l'association entre les deux variables, il existe toute une batterie de coefficients (**19**), dont certains, comme le _, sont dérivées du _^{2}, alors que d'autres sont des "mesures" purement descriptives sans fondement statistique solide. On précisera aussi que certains de ces coefficients mesurent l'intensité de la relation de façon symétrique (comment X et Y dépendent mutuellement l'un de l'autre), alors que d'autres sont utiles dans le cadre d'une question orientée: quelle est l'intensité de la dépendance de Y par rapport à X ? Quant à la prédiction de Y connaissant X, c'est la *probabilité conditionnelle* d'appartenance à une modalité de Y une fois la valeur de X connue.

Si les variables sont toutes qualitatives et que leur nombre est supérieur à deux, le tableau de contingence devient un cube (pour 3 variables) ou un hypercube (4 variables ou plus). La technique pour tester l'existence d'interdépendance entre les variables prises globalement est le G^2, dit *rapport de vraisemblance*. Au delà de ce test, le *modèle log-linéaire* constitue une façon de structurer les relations simples (bi-variables) ou complexes (interactions d'ordre 2 ou 3) qui peuvent exister dans un tel tableau, en faisant apparaître la part de variabilité associée à chacune; cette technique convient donc bien à une approche exploratoire. Proche du précédent par le mode de calcul, le *modèle logit* consiste à désigner une modalité de l'une des variables du tableau comme une variable d'*output* binaire dont il faut prédire la probabilité de réalisation - à partir de la connaissance des valeurs de modalités prises par les autres variables. Les références (**2**, **4**, **25**) décrivent toutes ces techniques.

Lorsque l'on a affaire à deux variables de natures différentes (l'une X qualitative, l'autre Y quantitative) alors on se trouve dans une situation *d'analyse de variance avec facteur unique non contrôlé*, les modalités de la variable qualitative jouant le rôle de niveaux du facteur. Le rapport des variances (variance résiduelle/ variance totale, au niveau de Y) peut servir à mesurer le degré de dépendance de Y par rapport à X. La prédiction de Y par X peut se concevoir comme une moyenne conditionnelle, alors que la prédiction de X (variable qualitative) par Y fait appel à la *régression logistique*, c'est-à-dire une régression munie d'une transformation logistique sur la sortie (*output*).

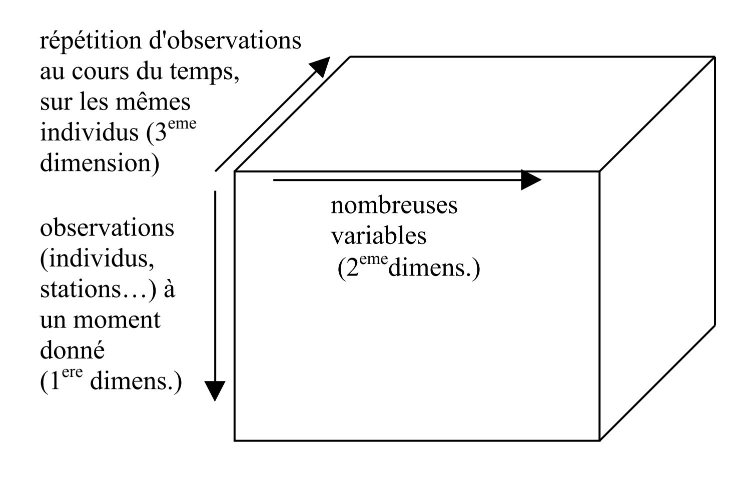
Dans le cas de figure le plus complexe, c'est-à-dire lorsque l'on a affaire à un certain nombre de variables (Y, $X_1 \ldots X_m$) de *différentes* natures et que l'on veut analyser la réponse de l'une d'entre elles (Y) aux valeurs prises par les autres, on aura intérêt à se replacer dans le cadre du *modèle linéaire général* (**21bis**). Il ne s'agit pas d'une technique particulière mais d'un modèle qui englobe, relie et généralise un certain bon nombre de techniques précédemment évoquées (régression linéaire, analyse de variance, régression multiple, modèle logit, régression logistique…) ainsi que d'autres techniques (*analyse de covariance*…). A toutes, il offre désormais une assise théorique plus solide et non assujettie à l'hypothèse de distribution gaussienne des erreurs (écarts entre les valeurs observées et les valeurs prédites). En effet, le critère d'ajustement et d'estimation employé par le modèle linéaire général est le critère du *maximum de vraisemblance*, et ce critère n'est pas aussi contraignant en matière de qualité des distributions - on montre d'ailleurs qu'il se réduit à un simple critère quadratique lorsque la distribution des erreurs est normale. Cependant, la difficulté d'écriture et de traitement de la fonction analytique dérivée du critère de maximum de vraisemblance oblige généralement à avoir recours à un algorithme d'optimisation "pas à pas" de type Gauss-Newton - ce qui ne constitue toutefois pas une difficulté réelle avec les moyens de calculs actuels.

2.3. Les techniques d'analyse multivariée.

Les analyses multivariées (ou *analyse de données "multidimensionnelle"*) trouvent leur justification dans le fait que les enquêtes ou les campagnes scientifiques, menées bien souvent dans une logique exploratoire et de façon pluridisciplinaire, conduisent à l'accumulation de *corpus* de données massifs, c'est-à-dire contenant de très nombreuses observations et variables. Face à cette situation, et plutôt que de réaliser un fastidieux travail d'analyse variable par variable ou par couple de variables comme ce que nous avons vu au § 2.2. précédent, on peut décider de traiter de nombreuses variables d'un seul coup en adoptant *une démarche exploratoire, à la recherche de grandes tendances* (que l'on pourra aussi appeler "structures principales de la variabilité") que l'on essaiera de matérialiser sous forme d'axes ou de 'types'. Précisons que les analyses multivariées font fi de la technique par laquelle les données ont été obtenues; elles s'appliquent indifféremment aux données issues de recensements, de sondages par quota, de sondages aléatoires ou de toute autre forme de collecte. Cette relative négligence vis à vis de la question de la représentativité des données les rend peu compatibles avec des objectifs d'estimations inférentielles et moins encore avec des démarches confirmatoires (test d'hypothèses). Mais là n'a jamais été leur ambition.

Les techniques d'analyse multivariée admettent généralement pour point de départ une table ou cube de données composé de deux ou trois dimensions: en ligne (verticale) on place les multiples réalisations d'observations nombreuses (individus enquêtés, stations observées, placettes visitées etc…), en colonne (deuxième dimension), on inscrit les valeurs des variables observées au niveau de chaque individu, en profondeur (troisième dimension), on place les répétitions éventuelles des visites aux mêmes objets (individus, stations…) au cours du temps.

Fig.1: rappel: notion de table ou cube de d'observations x variables



En partant des informations contenues dans un tel cube, il existe deux modes de traitements possibles, respectivement dénommés "groupement" et "ordination", qui dans tous les cas ont pour but de réduire l'information en une synthèse lisible.

2.3.1. Techniques de classification et typologie (**17**).

A partir d'un ensemble d'éléments observés (objets, individus, stations etc..) décrits par les mêmes variables, la démarche de *classification* (parfois appelé "groupement") consiste à faire émerger une partition de l'ensemble en un petit nombre de classes, en faisant en sorte que les éléments affectés à une même classe se ressemblent. Chaque classe pourra ensuite être décrite par le profil descriptif moyen des éléments lui appartenant, ce profil devenant un type ou *archétype*. C'est pourquoi on

parlera aussi de *typologie* pour qualifier ce genre d'approche. Du point de vue technique, les méthodes couramment employées pour faire apparaître ces classes et ces types sont la *classification ascendante hiérarchique* (C.A.H.) et les *nuées dynamiques*.

Dans tous les cas, il faut disposer au départ d'une *mesure de ressemblance* (ou de dissemblance) entre les éléments comparés deux à deux, et ensuite d'un *critère d'agrégation ou d'agglomération* qui sert à décider de la façon dont les éléments ou les groupes d'éléments vont se rassembler pour aboutir à la constitution des classes de la partition finale.

Pour mesurer la ressemblance entre individus ou stations observés, on utilisera généralement l'une des formules qui sert à mesurer l'association ou la corrélation entre deux variables sur un échantillon (coefficient de corrélation, coefficient d'association - cf. § 2.2.3), bien que le problème soit ici complètement différent. Pour l'étape suivante qui consiste à réaliser l'agglomération ou le groupement (au sens strict) des individus, il existe une méthode assez générale, décrite par Lance & Williams (**18bis**), que l'on retrouve implémentée dans la plupart des logiciels commercialisé.

2.3.2. Techniques d'ordination en espace réduit.
(référence générale: **19**)

Les techniques d'ordination cherchent en premier lieu à dégager des tendances de covariabilité entre les variables présentes dans un tableau (observations _ variables). De telles tendances peuvent se concrétiser sous forme de combinaisons linéaires associant des variables présentant entre elles de fortes affinités de co-variabilité (*i.e.* corrélation *lato sensu*). Chacune de ces combinaisons peut être considérée comme une nouvelle variable, artificielle, synthétique, dite "axe factorielle". Le premier de ces axes sera constitué par une combinaison d'un grand nombre de variables présentant de fortes affinités. Le second sera orthogonal au premier et s'appuiera sur d'autres co-variations impliquant généralement d'autres variables, et ainsi de suite… Finalement, on retiendra deux à cinq axes factoriels, définissant deux à deux des plans sur lesquels on pourra repositionner les éléments observés du tableau initial. Sur ce principe existent plusieurs techniques d'ordination, qui ont été développées pour répondre aux différentes natures possibles (qualitatives/ quantitatives) des variables d'origine, et donc aux différentes façons de mesurer les co-variabilités.

L'analyse en composante principale (ACP) est particulièrement adapté aux tableaux (observations x variables) dans lesquels les variables sont toutes quantitatives, à variations continues ou quasi-continues (mesures, comptages d'éléments très nombreux) (**20**).

L'*ACPVI* est un développement particulièrement de l'ACP destiné à traiter les cubes de données de type "observations x variables x temps". Elle permet notamment de bien extraire (sur un premier axe) les structures généralement très fortes associées à la saisonnalité, ce qui favorise ensuite une analyse claire des autres structures, souvent plus ténues (effet de l'environnement physique, variations interannuelles…).

L'analyse factorielle des correspondances (**1**) est adaptée aux tableaux (stations _ variables) où les variables sont des effectifs (par ex. des comptages d'espèces biologiques). Dans ce cas, la meilleure mesure de co-variation est de type _^{2} , et c'est à partir de cette mesure que la méthode forme les axes factoriels sur lesquels seront repositionnées les observations (stations).

L'analyse des correspondances multiples (ACM) est la méthode la plus puissante et la plus générale, car elle accepte tous types de variables dans le tableau de données (**1**). Elle nécessite la transformation préalable de chacune des variables en variable qualitative à deux, trois ou quatre

modalités, puis le calcul du tableau de Burt, c'est-à-dire le tableau des tableaux de contingences de toutes les variables (qualitatives) croisées deux à deux.

Les techniques d'analyses multivariées que nous venons de voir, ainsi que d'autres associées à des usages plus ou moins spécifiques (*mutidimensionnal scaling*, analyse symbolique…), tendent aujourd'hui à être utilisées en batteries plus ou moins automatisées, pour permettre l'attaque de corpus d'informations de plus en plus gros et hétéroclites. Cette philosophie très pragmatique de l'extraction de l'information par tous les moyens possibles a pris récemment le nom de *data mining*.

2.3.3. <u>Remarques générales sur les techniques d'analyse multivariée</u>.

Nous conclurons le domaine des analyses multivariées en relevant certaines difficultés de perception et d'usage qui leur sont associées. Ces difficultés proviennent d'une part de la relative complexité mathématique des calculs, d'autre part de l'excès de crédit qu'on a longtemps accordé à ces techniques quant à leur capacité à digérer et à synthétiser de façon correcte de grandes masses d'informations hétérogènes, voire brutes. Or, malgré leur prétention à un usage universel et malgré leur tolérance affichée par rapport à la nature des *inputs*, il faut reconnaître que les techniques d'analyse multivariée ne sont guère plus agiles que les autres quant il s'agit "d'avaler" directement des formats d'information complexes fournis par les systèmes d'enquête. En effet, ces formats se caractérisent généralement par des structures multi-niveaux, la visite à certaines unités d'observation étant subordonnée à la visite préalable d'unités englobantes par rapport aux précédentes (par exemple: un ménage n'est enquêté qu'après la visite et le recueil d'information sur le village où il réside). Et l'analyse ne peut être réussie que si elle parvient à prendre en compte conjointement les informations (*i.e.* les variables) issues de différents niveaux (**10bis**). Par exemple, on ne peut réussir une bonne ordination ou une bonne typologie du comportement économique des ménages sans intégrer des informations relatives à leurs agglomérations de résidence. Pour y parvenir, il faut d'abord maîtriser quelques problèmes informatiques (mais ceci ne pose aujourd'hui guère de problème avec la méthodologie des bases de données relationnelles) et, dans un second temps, résoudre des problèmes d'analyse statistique et de présentation des résultats. Les logiciels commerciaux d'analyse statistique multivariée ne traitent pas ces situations de façon simple et transparente.

2.4. <u>L'analyse de l'information observée le long d'une dimension simple</u>
(par exemple le temps ou bien un axe dans l'espace)

2.4.1. Analyse de séries de données d'observations successives discrètes, acquises selon un pas de temps fixe le long d'un axe (références générales: **14**, **8**).

On considère que l'on a affaire à une série longue (par le nombre d'observations successives) concernant une ou plusieurs variables. C'est par exemple sous un tel format que se présentent les données des observatoires hydrologiques ou météorologiques. Il existe alors deux formes d'étude statistique: l'une consiste à analyser les variations de chaque variable considérée isolément (approche monovariée), l'autre à étudier les relations qui existent entre les variations de plusieurs variables (approche multivariée). Les deux démarches ne sont bien sûr pas incompatibles, la première constituant souvent un préliminaire nécessaire à une bonne mise en œuvre de la seconde.

a) approche monovariée

On considère comme exemple une série temporelle de 3 ans, constituée de 36 observations mensuelles successives d'une variable quantitative, et ayant l'allure suivante (fig.2):

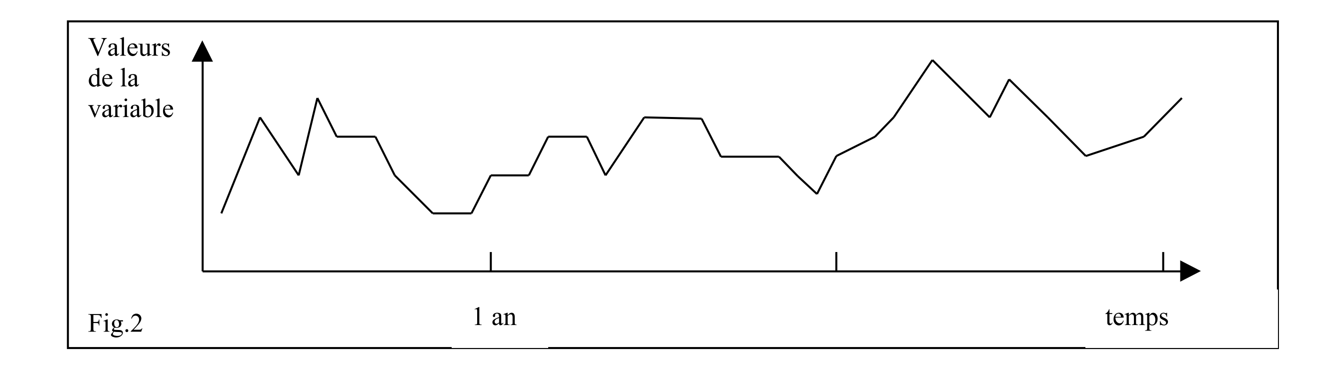


Fig.2

Celui qui veut décrire et analyser une telle série va rencontrer plusieurs problèmes et questions:

- Comment éliminer, le cas échéant, le *bruit*, c'est-à-dire les variations aléatoires à hautes fréquences qui gênent la lisibilité de la série. C'est le problème du *filtrage*, aujourd'hui très bien résolu (**24**) par des approches plus ou moins sophistiquées - les plus performantes, comme la *décomposition en ondelettes*, proviennent des recherches en matière de traitement du signal. Cependant, dans le domaine des données environnementales, on pourra généralement se contenter d'une simple moyenne mobile portant sur quelques termes, éventuellement pondérée (par ex.: 1/4, 1/2, 1/4).

- Comment tester l'existence d'une tendance monotone, comment la représenter, et éventuellement l'extraire. C'est un problème qui se pose couramment dans le domaine environnemental (ex.: telle série séculaire de température permet-elle de conclure que la région se réchauffe ?). Le test de *corrélation de rang de Kendall* (où l'on teste la corrélation entre le rang d'occurrence dans le temps et le rang de classement par valeurs - **18**) est approprié pour détecter l'existence de tendances monotones, que celles-ci soient linéaires ou non. Précisons qu'il est préférable de faire le test sur la série brute, non filtrée. Une fois l'existence de la tendance avérée, on peut chercher à la matérialiser en ajustant un modèle linéaire de degré un ou deux (de type régression Y= f(t)). Les valeurs Y prédites par le modèle peuvent alors être ôtées aux valeurs Y observées, ce qui permet théoriquement de se retrouver avec une série stabilisée.

- Comment rechercher et faire apparaître une composante de variation périodique (typiquement: une saisonnalité) et, le cas échéant, la retirer ? Il existe plusieurs techniques efficaces, à condition que la périodicité T de la variation soit fixe. Le *corrélogramme* (**19**) consiste à corréler la série décalée avec elle-même (non décalée), jusqu'à trouver le décalage qui maximise la corrélation: on détecte ainsi la périodicité principale. De façon analogue, le *périodogramme de contingence* détecte une périodicité de récurrence sur une série d'états qualtitatifs. La *table de Buys-Ballot* est une technique qui permet de faire apparaître la forme moyenne de la variation périodique, et donc de la retirer par la suite. Le *périodogramme* est une extension de la technique précédente (**19**).

b) approche multivariée.

En approche multivariée, l'étude des séries temporelles conduit généralement à un problème de prédiction. Mais deux cas peuvent alors se présenter:

- le cas où il n'y qu'une seule variable à prédire. Concrètement, il s'agit d'identifier et d'estimer un modèle linéaire qui réalisent à chaque temps t une prédiction de la valeur y_t prise par la variable Y,

en fonction des valeurs observées aux pas de temps précédents (t-1, t-2…) pour cette même variable Y et, éventuellement, en fonction des valeurs observées pour une ou plusieurs autres variables dites "explicatives" (U, X… Z). Plusieurs problèmes surgissent à ce niveau et rendent l'estimation du modèle difficile: le premier est constitué par le fait que les valeurs successives d'une même variable ne sont pas, en général, indépendantes. Le second est constitué par l'explosion du nombre de paramètres qui résultent de la combinatoire des effets à prendre en compte (n variables explicatives _ p retards). Ces difficultés sont traités et largement résolus par la technique ARMA (*autoregressive and moving average model*) qui permet d'estimer un modèle linéaire à la fois efficace (par ses capacités de prédiction), parcimonieux et stable (**8**).

- le cas où il y a un vecteur de variables à prédire en fonction des observations passées de ces mêmes variables (voire même d'autres variables, dîtes "externes" ou "forçantes"). Ce problème est celui des *modèles à équations multiples*. Il est très vaste et largement ouvert sur une autre discipline: la modélisation dynamique. En effet, les *systèmes linéaires dynamiques* de type *Lotka-Volterra prédateur-proie* ou de type *modèle démographique de Leslie* (**21**) peuvent, s'ils sont exprimés en temps discrets (c'est-à-dire sous la forme d'équations aux différences finies), être abordés dans le cadre général du modèle linéaire, du moins en ce qui concerne l'étape de l'estimation de leurs paramètres. Dans le cas particulier où les variables sont qualitatives, le vecteur observé est un vecteur d'états qu'il faut pouvoir prédire en fonction des vecteurs d'états précédemment réalisés: ce problème particulier est traité dans la cadre de la *théorie des graphes et des processus stochastiques discrets* (**3**) avec un cas singulier et simple dénommé *chaîne de Markov* dans laquelle la mémoire du système est limité à une unité de pas de temps (*i.e.* les états au temps t ne sont influencées que par les états au temps t-1 mais pas par les états précédents). L'estimation des paramètres de probabilité de transition d'états dans une chaîne de Markov est un problème qui peut également être abordé dans le cadre du modèle linéaire.

2.4.2. Analyse de la position d'événements pouvant survenir à tout instant dans un temps continu vécu par des individus identifiés (**6**).

Contrairement à ce qui est décrit au § précédent (2.4.1), on considère ici un temps continu. Au cours de ce temps peuvent survenir un petit nombre d'événements bien particuliers. Le cas le plus typique est celui de la vie d'un individu humain: à partir de t_0 (la naissance), surviendront à des dates t_i des événements de $types_i$ obligatoires ou facultatifs comme le premier emploi, le mariage, le premier enfant, le deuxième enfant, le divorce, la survenue d'une maladie, la mort… On peut considérer ces dates t_i comme des variables intéressantes à analyser, de même que les écarts entre plusieurs dates t_i (par exemple l'écart entre le mariage et la naissance du premier enfant, ou entre le début d'une maladie et la mort). On peut aussi faire intervenir des variables externes, environnementales, connues en parallèle sous une forme de série temporelle classique (cf. § précédent), et il s'agira alors d'évaluer si ces variables agissent sur la survenue des événements décrits plus hauts. En *input* de telles analyses, on devra disposer des informations sur la vie ou le parcours de nombreux individus, avec, pour chacun d'entre eux, les dates d'occurrence des événements étudiés. Ces analyses sont très utilisées en démographie et socio-économie (on parle alors *d'analyse de biographie*), en épidémiologie et en recherche clinique (où on les rencontre sous les appellations *d'analyses de survie*, *d'analyses de cohortes*…). Réputées difficiles du fait des problèmes d'interdépendances entre événements successifs (on ne peut pas divorcer avant de s'être marié !), ces analyses peuvent être menées à bien à l'aide de modèles de distribution appliqués aux processus temporels continus ou bien à l'aide de différentes techniques de la famille du modèle linéaire.

2.5. <u>Analyse spatiale: observations positionnées dans un espace bi-dimensionnel</u>.

(référence générale: **7**).

2.5.1. Recherche de tendances, régularisation et interpolation.

On dispose de valeurs d'une variable quantitative Z observée sur un semis de points régulier ou irrégulier dans un espace bidimensionnel doté de coordonnées orthogonales x, y (de type latitude, longitude). On se propose d'avoir un modèle qui fournit en tous points une prédiction de Z en fonction des coordonnées (x, y) et en fonction des valeurs de Z observées dans un voisinage plus ou moins lointain. Cela permettra notamment d'obtenir une grille régulière et dense de valeurs Z prédites, et ensuite, à partir de cette grille, d'effectuer des estimations de moyennes ou de totaux régionaux, ou bien de tracer des isolignes pour établir des cartes etc… Ce type de problème peut être abordé avec deux techniques:

- la technique dite "des *surfaces de tendance*": elle consiste à ajuster un modèle où la variable Z sera définie comme une fonction polynômiale en x et x, incluant des termes d'interactions de type xy et des formes quadratiques, voire cubiques, comme x^2 ou x^3. Les valeurs z observées servent à l'identification du modèle et à l'estimation des paramètres du polynôme. Il ne reste plus alors qu'à prédire Z en tout point (x, y), ce qui permet ensuite de tracer des isolignes. La faiblesse de cette méthode est que toutes les valeurs z observées influencent avec la même intensité les valeurs z prédites dans toutes les parties de l'espace, et ce quelques soient leurs positions. Il s'ensuit généralement des prédictions (*i.e.* des extrapolations) aberrantes dans certaines parties de l'espace.

- le *krigeage* (**22**) est une technique plus souple: les valeurs z sont prédites en chaque point (x, y) par un modèle linéaire qui utilisent les valeurs voisines avec des pondérations liées à leur distance et à leur orientation par rapport à la variable à prédire.

2.5.2. Modélisation de la répartition d'objets dans un espace continu bidimensionnel: processus de points. (référence: **23bis**)

De façon analogue à ce qui est décrit au § 2.4.2, on considère ici un espace continu et l'on s'intéresse à la position - dans cet espace - d'objets appartenant à une ou plusieurs catégories: ces objets sont-ils dispersés au hasard, surdispersés ou agrégés ? quelle allure peut prendre la distribution des distances entre objets voisins ? y-a-t'il des tendances à des co-occurrences, à des voisinages ou au contraire à des répulsions entre les objets appartenant à deux catégories différentes ? Ces questions apparemment simples sont généralement complexes à traiter sur le plan mathématique et font appel aux aspects avancées de la théorie statistique.

CONCLUSION

Les hypothèses restrictives qui conditionnent l'usage de la plupart des techniques statistiques sont souvent très sévères en regard des caractéristiques réelles des lots de données disponibles. Or, ceci n'arrête pas les utilisateurs qui, lancés à la poursuite de leurs objectifs scientifiques, franchissent allègrement les limites associées aux conditions de validité des techniques. Plutôt que de lutter contre ces infractions en s'érigeant en gendarmes, nombre de statisticiens préfèrent travailler à l'élargissement de l'assise mathématique des techniques statistiques, en essayant de traiter les cas (fort nombreux !) où, pour différentes raisons, la convergence vers une distribution limite simple ne se produit pas. Quelles sont les distributions que l'on peut alors obtenir, et comment les décrire en termes mathématiques ? Et comment alors effectuer des estimations, des prédictions, des tests? C'est là le principal moteur de la recherche en statistique.

Références bibliographiques des numéros cités.

1- Benzecri, J.P. et coll. (1973): L'analyse des données. Tome 1: la Taxinomie. Tome 2: l'Analyse des correspondances. Dunod. Paris.

2- Bishop Y.M.M., E. Fienberg et P.W. Holland (1975): Discrete Multivariate Analysis: Theory and Practice. MIT Press. Cambridge, Massachusetts. 557 p.

3-Chrétienne Ph. et Faure R. (1974): Processus stochastiques, leurs graphes, leurs usages. Coll. Programmation. Gauthier-Villars. Paris. 132 p.

4-Christensen R. (1990): Log-linear models. Springer Verlag. 408 pp.

5-Cochran W.G. (1977): Sampling techniques. 3eme édition. J.Wiley and Sons. New York.

6-Cox D.R. et Oakes D. (1984): Analysis of Survival Data. Chapman & Hall. London.

7-Cressie, N.A.C. (1991): Statistics for spatial data. John Wiley & Sons. New-York.

8-Diggle, P.J. (1990): Time series: a biostatistical introduction. Oxford University Press. Oxford.

9-Draper N.R. et H. Smith (1981): Applied Regression Analysis. Second Editions. John Wiley 1 Sons. New-York.

10-Droesbeke J.J., B. Fichet et P. Tassi eds. (1987): Les sondages. Coll. ASU. Economica. 310 pp.

10bis: Dubois J.L. et Blaizeau D. (1989): Connaître les conditions de vie des ménages dans les pays en développement. Tome 1: Concevoir l'enquête. Ministère de la Coopération et du Développement. 165 pp.

11-Efron B. et Tibshirani R.J. (1993): An introduction to the bootstrap. Collec. Monographs on Statistics and Applied Probability 57. New-York. 446 p. Chapman & Hall.

12- Frontier (1981): Méthode statistique. Collec. Abrégés. Masson. 246 p.

13- Frontier (1982): Stratégie d'échantillonnage en écologie. Masson. Paris.

14- Gourieroux C. et Monfort A. (1983): Cours de séries temporelles. Economica. Paris. 380 p.

15- Grangé D. et L. Lebart eds. sc. (1994): Traitement statistique des enquêtes. Dunod. Paris. 255 p.

16- Hertz J., A. Krogh et R.G. Palmer (1991): Introduction to the Theory of Neural Computation. Redwood City, CA: Addison-Wesley.

17- Kaufman L. et P.J. Rousseeuw (1990): Findings group data. An introduction to cluster analysis. John Wiley and Sons. New-York.

18- Kendall M.G. (1948): Rank correlation methods. C. Griffin & Co, London, 160 p.

18bis - Lance G.N et Williams W.T. (1967): A general theory of classificatory sorting strategies. *Computer Journal*, 10: 271-277.

19- Legendre L et P. Legendre (1979): Ecologie numérique. Tome I: Le traitement multiple des données écologiques. Tome 2: La structure des données écologiques. Masson. Paris.

20- Lefebvre J. (1976): Introduction aux analyses statistiques multidimensionnelles. Masson. Paris. 219 p.

21- Leslie P.H. (1945): On the use of matrices in certain population mathematics. Biometrika, 33, 183-212.

21bis- McCullagh P. et J.A. Nedler (1989): Generalized Linear Models. Chapman & Hall. London.

22- Matheron G. (1973): The intrinsic random functions and their applications. Advances in Applied Probability, 5, 439-468.

23- Powell T.M. et Steele J.H. (1995): Ecological Time Series. Chapman & Hall. 491 pp.

23bis - Ripley B.D. (1988): Statistical Inference for Spatial Processes. Cambridge University Press.

24- Salman W.P. et Solotareff M.S. (1982): Le filtrage numérique. Eyrolles. Paris. 174 p.

25- Santner T.J. et Duffy D.E. (1989): The Statistical Analysis of Discrete Data. New-York. 367 pp.

26- Scherrer B. (1984): Biostatistique. Gaëtan Morin éditeur. 850 p. Québec.

27- Tomassone R., E. Lesquoy et C. Millier (1983): La régression. Nouveaux regards sur une ancienne méthode statistique. Coll. INRA, 13. Masson. Paris. 180 p.

1.3 Symbolic Representation

This section deals with a state of the art of the basic Symbolic Representation techniques.

A survey of symbolic knowledge representation

Aurélien Slodzian
VUB Artificial Intelligence Laboratory
aurelien.slodzian@vub.ac.be

September 10, 1999

Abstract

This document presents a summary survey of the most fundamental techniques in symbolic representation. We will rather focus on *modern* representation frameworks rather than on *recent*ones, which often put new clothes on very old principles. Our aim is rather to explain the principles rather than listing all the implementations and syntactic variation of the representation methods.[1]

Contents

[1]This paper a deliverable of the SIMES project, funded by the EC in the context of the INCO-DC Program, as project number 961620.

1 Introduction: from data to knowledge

Todays computers have seen their computing power increase exponentially, as well as their storage capacity. The conjunction of these very pragmatic factors opened the way towards huger application software that make use of a proportional amount of information. In the context of domain specific applications – which is the case of the SIMES project – the problem is therefore not anymore to optimize the storage of the information but rather to provide tools to structure and organize great amounts of knowledge it in the most rational way.

Since the 1950s, Artificial Intelligence (AI) – a subfield of computer science – has focused, among others, on the problems of knowledge representation. Researchers in AI have initially addressed this issue with the aim to provide support for more complex inferences, but, as time passed, their results proved to be of the greatest importance for the organization of large knowledge bases. Indeed, rather than core techniques, they propose logical frameworks to organize the data in a way which is consistent with the conceptual structure of the domain.

The problem of representing knowledge may not be abstractly separated from the further usage of this knowledge. Not only the representation must be suited for the particular algorithms that will be applied to it, but also the type of knowledge, its conceptual organization depend ultimately on the inferences which will be drawn.

We will therefore place ourselves in the context of a artificial system design and assume we have a number of functionalities to reach. We can look on the designed system from three perspectives, which form as many *description levels*. The most abstract description level, the *knowledge level*, considers this system as a rational agent to which a number of tasks are ascribed and which tries to achieve them by applying rationally some methods to the available knowledge. Less abstract is the *symbol level*, at which the system is considered in a mechanistic way, as functions that operate over a symbolic representation of the knowledge. Finally, *sub-symbolic* representations will focus on the internal mechanics of the system, without considering its goals.

These notions deserve a few words of explanation.

The most common practice in computer science is to write programs. From the above point of view, this consists in describing a system by defining functions that operate over symbols which refer to some "real world". Which symbols refer to what is a convention decided by the programmer and this relationship is generally not written down, excepted, maybe, as comments. This is exactly what is called a symbol-level description of the software system.

When this description – the program – is compiled and put into

operation, we end up with a stream of bytes into a computer memory. This is a sub-symbolic representation of the same system. It is clear that this representation looses track of any symbolic relationships between computer state and the real world.

Symbol-level techniques were a great progress since they allowed system developers to specify the behavior of a system in symbolic, more abstract, terms. However, the symbol-level description of a system does neither reflect its rationale, nor its semantics. In other words, the representation rules are not made explicit.

This lead, about two decades ago, Newell [10] to introduce a new approach, which he called the *knowledge-level.* Newell considered systems composed of agents, in the sense of entities that have a particular behavior and interact with their environment. He furthermore proposed to assume that these agents would be *rational* and would behave as if they had some knowledge about the world and used this knowledge rationally to reach their goals. From this starting point, different methodologies were elaborated to put Newell ideas into practice, among which are KADS [15, 7] and the componential methodology [13]. These methodologies have all in common to describe systems in terms of goals, tasks, abstract methods and knowledge. The resulting descriptions might be considered as very abstract programs that would express the semantics of the system, and which require, of course, much more complex interpreters.

The knowledge-level approach brings a new perspective on symbolic data representation: it considers the semantic structure of the knowledge and adapts its concrete representation accordingly. To mark this difference, the term *knowledge representation* is used, and the design of systems is called *knowledge-level modeling.*

While developing new techniques, AI researchers invented many knowledge-representation techniques,Most of them being, in a way or an other, derivatives of a few fundamental ones, which are described in the next section.

2 Classical representation techniques

A variety of ways of representing knowledge have been exploited in A.I. programs. But before we can talk about them individually, we must keep in mind that we are dealing with two different kinds of entities:

- Facts are the things we want to represent.
- Representations of facts – in some chosen formalism – are the things we will actually be able to manipulate.

The issue in representing knowledge is to find a balance between simpler formalisms and simpler relationship between the facts and their representation. Indeed, a simpler representation facilitates the manipulation of the symbols, but reduces the expressive power of the chosen formalism, to the point that it may introduce important discrepancies. The most often encountered problem with too simple formalisms is that the designer's attention is more focused on the limitations of the representation than on the real problem he has too solve. This is the case, for example, with database systems which impose to think in terms of records and fields, while it is seldom a "natural" way people manipulate knowledge.

This being said, the following subsections will present the most fundamental knowledge-representation frameworks.

- *Logic* represents knowledge in a predicative way;
- *Semantic networks* organize assertions in a network of concepts;
- *Conceptual graphs* organize object descriptions in a graph of relations;
- *Production systems* describe rules that allow computers to make inferences;
- *Frames* consider the domain as a collection of hierarchically classified objects which may be described by a number of their features.
- *Ontologies* define the vocabulary of a domain.

2.1 Logic

Logic was one of the first representation scheme used in AI. It has two important and interlocking branches: the first is the consideration of the relations and implications one can formalize about the particular domain of knowledge being considered, and the second is the deductive structure that determines what can be inferred if certain axioms are taken to be true. Logic is concerned with the form, or *syntax* of statements and with the determination of truth by *syntactic* manipulation of formulas.

Most logic representation formalisms are "computerized" versions of first-order logic, i.e. of predicate calculus.

- A set of *symbols* is defined that each represent entities of the world. These entities may refer to objects (e.g. a car) as well as to properties of these objects (a.g. a color).

- Some special symbols are taken as *variables* that may represent any object in a range (possibly all objects, but generally a subset of objects of a given category).
- Symbols may appear as arguments of *predicates*, which represent statements about entities. Predicates are usually represented in a functional style.

In this document, we will use the KIF [5] representation of logic. This is a prefixed notation where predicates are represented as lists, the first term of which is a symbol identifying the predicate and the remaining ones the arguments of the predicate.

For example, the symbol `John` might well refer to a real-world entity called "John". The relationship between the symbol and the real object lies in the mind of the writer (and hopefully of the reader also) and may by no means be represented formally.

Then we may represent an assertion about the object called "John", and, for example, state that "John is a man":

```
(man John)
```

Again, the notion of manhood is not represented formally and, from a "pure theoretical" point of view, the previous expression only states something like: "there is a predicate – `man` – which may be applied to the object identified as `john`".

Logic allows to represent much more complex statements. The following one says in substance: "all men are animals".

```
(forall (?x) (=> (man ?x) (animal ?x)))
```

It is the way computer logicians represent a formal assertion that would otherwise be expressed by mathematicians as:

$$\forall(x) man(x) \Rightarrow animal(x)$$

The awful ASCII representation was imposed by limitations of computers that did not allow – at that time – to use mathematical symbols.

These examples show that logic representation relies on a correct interpretation of expressions with respect to the real world they are supposed to model. It is important to note that there is not a unique way to represent statements. Firstly, there might be syntactical variants, depending on the language inventor. Secondly, and more importantly, the interpretation depends on the concepts that are identified as elements of the logic.

For example, one might consider that "red" is a possible property of cars. In which case, a red car would be represented as follows:

```
(red car)
```

But one might choose to consider rather a "color" predicate, with two arguments: the object and its color. The previous expression would then be re-formalized as follows:

```
(color car red)
```

For those who prefer a more functional style, we might consider that "color" is a function that may be applied to objects, and which value is a symbol that refers to the real color of the real object.

```
(= (color car) red)
```

To cope with those syntactic variations, one has to introduce meta-rules that describe the relationships between the different representations.

```
(forall (?x) (<=> (red ?x) (color ?x red)))
```

$$(\forall x) red(x) \Leftrightarrow color(x, red)$$

```
(forall (?x ?y) (<=> (color ?x ?y) (= (color ?x) ?y)))
```

$$(\forall x)(\forall y) color(x, red) \Leftrightarrow color(x) = y$$

This shows how complex may become a logical representation if all syntactical variants are to be allowed. But the real drawback appears with the question of the inferences that can be made.

The major disadvantage of logic comes from the separation of *representation* and *processing*. The difficulty is to determine how to use the facts stored in the system's data structures rather than in deciding how to store them. Thus, separating the two aspects and concentrating on epistemological questions merely postpone addressing the problem. Indeed, our goal is not only to *represent* facts and assertions, but to allow computers to derive their logical *consequences* and, since Godel, we know that this is not possible in a complete manner. However, attempts were made in this direction, among which are EPILOG, a KIF based inference engine (Stanford University Logic Group) and COQ, a proof assistant (INRIA).

Despite this problem, logic representation brings important benefits:

1. Logic is often a *natural way* to express certain notions. The expression of a problem in logic often corresponds to our intuitive understanding of the domain. Logical representation is also easier to reformulate; thus experimentation is made easier.

2. Logic is *precise*. There are standard methods of determining the *meaning* of an expression is a logical formalism.

3. Logic is *flexible*. Since logic makes no commitment to the kinds of process that will actually make deductions, a particular fact can be represented in a single way, without having to consider its possible use.

4. Logic is *modular*. Logical assertions can be entered in a database independently of each other; knowledge can grow incrementally, as new facts are discovered and added.

Logic is the "mother" of all representation techniques in AI. It is the one that imposes less structure on the knowledge, sticking more to its semantics. As we said before, this has the consequence to give it more representational power.

Other representation schemes are to some extent derivatives of the logic representation and they are each characterized – independently of syntactic variations – by the structure on the knowledge.

2.2 Semantic Networks

Semantic networks form a group of representation formalisms that share a common notation consisting of *nodes* (drawn as boxes in the illustrations) and *arcs* (or *links*; drawn as arrows) connecting the nodes. Both the nodes and the arcs can have labels. Nodes usually represent *objects*, *concepts*, or *situations* in the domain, and the arcs represent the relations between them.

The superficial similarity of this notation is all that most semantic network systems have in common. Semantic networks were first developed with the aim to represent the meanings of natural language sentences in terms of objects and relationships among them. However, some researchers in psychology have developed semantic networks as psychological models of human memory and researchers in computer science have been more concerned with developing functional representations for the variety of types of knowledge needed in their systems. Because of these diverse goals, there is no simple set of unifying principles to apply across all semantic network systems.

Certain themes, however, are common to most versions, such as:

- *nodes* in the net represent concepts of entities, attributes, events, and states;
- different nodes of the same concept type refer to different individuals of that type, unless they are marked with a name, identifier, or coreference link to indicate the same individual;
- *arcs* in the net, called *conceptual relations*, represent relationships that hold between the concept nodes (labels on the arc specify the relation types);

- some conceptual relations represent linguistic cases, such as agent, object, recipient, and instrument (others represent spatial, temporal, logical or inter-sentential connectives);
- concept types are organized in a *hierarchy* according to levels of generality, such as “Entity”, “Living-Thing”, “Animal”, “Carnivore”, “Feline”, “Cat”;
- relationships that hold for all concepts of a given type are inherited through the hierarchy of subtypes.

Besides these commonalities, the various networks diverge on a number of issues: philosophical questions of meaning; methods for representing all the quantifiers and operators of symbolic logic; techniques for manipulating the networks and drawing inferences; and notations and terminology that differ from one author to another. Despite the differences, all the versions are based on some common assumptions: network notations are easy for people to read, efficient for computers to process, and powerful enough to represent the semantics of natural languages.

Semantic networks can be easily represented graphically, which is one of their advantages. Figure 1 represents graphically the sentence “A dog is greedily eating a bone.” A textual representation of the same might be, for example:

```
(eat (agent dog) (manner greed))
```

.

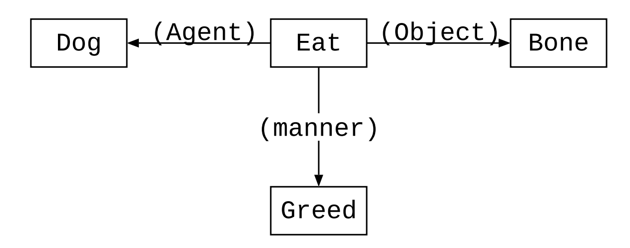


Figure 1: A sample semantic network

The textual representation is meant for the computer to make inferences and the graphical one for human eyes only.

2.3 Conceptual graphs

Conceptual graphs are a derivative of semantic networks. Their difference is mainly that the nodes are usually reserved for objects and arcs for verbs and properties.

Conceptual graphs are therefore more oriented towards the representation of knowledge structure and are not suited for representing assertions, like general semantic nets do. The example in figure 1 above could therefore not be represented as such.

However, since the structure of conceptual graphs is simpler they may be more efficiently used than semantic nets. Again, we see that simpler structures provide more efficiency at the cost of representation power.

Figure 2 shows a very simple conceptual graph (usual ones have hundreds of nodes). We see objects (represented inside squares) and relations (arrows) among them. What does not appear on the graph is that conceptual graphs systems generally allow to define inheritance rules, as well as relation definitions.

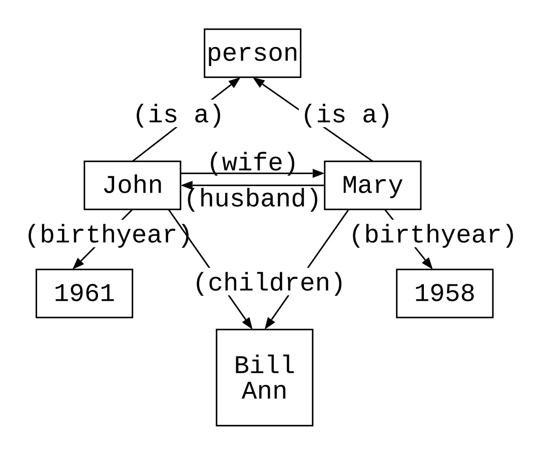


Figure 2: A sample conceptual graph

Using the KRS [8] representation syntax, the nodes might be represented as follows:

```
(defconcept JOHN
   (a person
      (sex male)
      (wife MARY)
      (children (list BILL ANN))))

(defconcept MARY
   (a person
      (sex female)
      (birthyear [number 1958])))
```

This has to be completed with the definition of the notion of `person`, which the following expression does. The object `person` is

declared to have two properties: its `birthyear`, which is a number, and its `age` which is also a number. The second property may furthermore be computed by providing its *definition*.

```
(defconcept PERSON
  (birthyear (a number))
  (age (a number
          (definition [form (- (>> current-year)
                               (>> birthyear))])))
  (children (a list
               (definition
                 [form (>> children of partner)]))))
```

For those who are familiar with object-oriented programming, it is important to note that the concept of `person` is used as a prototype for `john` and `mary`. This is a different notion of class and inheritance, which we will not discuss here. The point is that any object may be used as a prototype for any other one. The reader should refer to object-oriented literature for more information on this subject.

Finally, an important feature of conceptual graphs is that they also allow the definition of relations. In the last example, the `children` of a `person` are defined to be the same as the children of his/her `partner`. The `partner` relation has then to be defined as either the `wife` or the `husband`, depending on the `sex` of the person. Similarly, the `husband` relation might be defined as the symmetric of the `wife` relation, so that when, in the code above, `john` is declared to have `mary` as `wife`, then the system may deduce that `mary` has `john` as husband, and that they are thus `partners` of each other.

These examples show a very little of the power of conceptual graphs. But they also show their limitation: compared to logic and semantic nets, conceptual graphs have less expressive power. Just imagine how complex it would become to express that the children of the wife are not necessarily the children of the husband, or that persons might divorce ! Reciprocally, if one make the effort to express these, then he/she will get the deductive power of the conceptual graph engine. With a logic representation, nothing guarantees that the existing inference engines will be able to cope with the logic representation of the same notions.

2.4 Production systems

Production systems were first proposed by Post (1943) but have since undergone such theoretical and application-oriented development in AI that the current systems have little in common with POST's formulation. In fact, just as the term *semantic net* refers to several different

knowledge representation schemes based on the node-and-link formalism, so the term production system is used to describe several different systems based on the very general, underlying idea – the notion of condition-action pairs, called *production rules*, or just *productions*.

Since many *expert systems* were built on top of production systems, the two terms are often confused.

A production system consists of three parts: (a) a *rule base* composed of a set of production rules; (b) a special data structure which we shall call the *context*; and (c) an *interpreter*, which controls the system's activity.

A *production rule* is a statement cast in the form "IF this *condition* holds, THEN this *action* is appropriate.".

The *condition* part of a production is also called it's *IF part* or *left-hand side*. It states the conditions that must be present for the production to be applicable. The *action* part – also called the *THEN part*, or the *right-hand side* – specifies the appropriate action to take. During the execution of the production system, a production whose condition part is satisfied can *fire*, that is, have its action part executed by the interpreter. Typical production systems contain hundreds of productions in their rule base.

The *context* – or *working memory* – is the focus of attention of the production rules. The left–hand side of each production in the rule base represents a condition that must be present *in the context* before the condition can fire. Reciprocally, the actions of the production rules can change the *context*, so that other rules will have their condition parts satisfied.

Finally, there is the *interpreter*, which, like the interpreters in all computer systems, is a program whose job is to decide what to do next. In a production system, the interpreter has the special task of deciding which production to fire next. The interpreter cycle is summarized in figure 3.

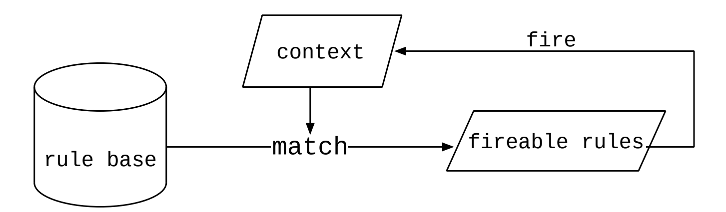


Figure 3: The interpretation cycle

Rules may be used to represent declarative as well as imperative knowledge.

- *Declarative rules*, in general, describe the way things work in the world. For example "*IF a patient has the flu, THEN the patient tends to exhibit fever and a runny nose.*"
- *Imperative rules* prescribe the heuristic methods that the knowledge system should employ in its own operations. For example "*IF you observe a patient with fever and a runny nose, THEN you should suspect that the patient has the flu.*"

The type of rules is not specifically correlated with the goal of the system, and both types of rules may be mixed in a single rule base. However, declarative rules are better suited for modeling domain knowledge since they keep system designer away from worrying about the knowledge base management. Imperative rules are rather used to control explicitly the behavior of the interpreter.

A lot of work has been done to build sophisticated interpreters that care about this issue. They are based on two inference mechanisms, namely *forward chaining* and *backward chaining*. The former derives new facts by recording the *THEN* part of rules in the context, until interesting facts appear that are notified to the user. It is used to look for the consequences of a number of assumptions. Reciprocally, backward chaining is used to determine the necessary conditions for a fact to be true. This is achieved by analyzing the right-hand sides of rules and gathering the left-hand parts, i.e. the conditions, that would produce the expected conclusion. When repeated recursively, this mechanism will effectively end-up with a number of conditions that, if taken as assumptions of a forward-chaining mechanisms, would bring to the expected conclusion. Backward-chaining is mostly used either for *proving* assertions (i.e. check if the discovered conditions are effectively met) or for *advising* actions (i.e. the system advises its user to act so as to meet the gathered conditions).

Rule-based systems have been – and remain – an important part of the research in AI. Their main problem is that the system designers have anyway to design many specific rules that control the rule-based system itself as soon as it becomes too large to be handled with a single rule base.

2.5 Frames

The notion of *frame* was introduced by Minsky [9] in 1975 as a basis for understanding visual perception, natural languages dialogues, and other complex behaviors. From a conceptual point of view, frames rely on an associationistic perspective on knowledge – that is, one thing always leads to another –, the units of knowledge representation be-

coming cohesive clusters of related facts rather than simple elementary facts.

Typically, a frame describes a class of objects. It consists of a collection of *slots* that describe aspects of the objects. These slots are filled by other frames describing other objects. The frames therefore form a network, connected through the slots. In this sense, frame based systems are relatively close to semantic nets and conceptual graphs. Note by the way that, like for all other representation schemes presented in this document, there are many frame systems, with many different features and syntax.

But the syntactic aspect of frames should not hide their initial intention: capture knowledge evolution and refinements. Indeed, a slot, considered as an aspect of an object or a situation, is not filled with a single value, like in classical attribute/value object systems. Instead, it has itself *facets*, which describes its different aspects. Therefore frame systems incorporate two levels of representation: representation of the "real world" facts, and information about the representation properties.

Let's take the marital example of section 2.3. The frame approach would be to first define a class (in fact rather a prototype), with a number of slots like `birthyear`, `age`, `children` etc. Each slot has different facets. For example, a facet of `children` is that when a new child is added, is should also be added to the `partner`. In the example below, this facet is identified by the `if-added` keyword.

```
(frame
  (name person)
  (partner ...)
  (children
     (value ()) ; default value = empty
     (if-added
        (push (newvalue)
              (>> children of partner)))))
```

Therefore, associated with each slot may be a set properties of that slot. Among them are the conditions that must be met by any filler for it; a default value, so that, in the absence of specific information to the contrary, things can be assumed to be as they usually are; or procedures that are to be executed when such or such event arises – like, for example, the modification, of the value of the slot. The use of such procedures embedded in an otherwise declarative structure is called *procedural attachment*.

From a technical point of view, frames are a very powerful generalization of the object oriented approach. Like objects, frames have attributes (slots). But unlike classical objects, frame slot are not re-

stricted to one value. Like object classes, frames describe typical situations which may be inherited. However, frame systems allow to control the inheritance mechanism itself, especially in the case of multiple inheritance. Indeed, it is possible, with specific slots, to define from which class the default value of a slot should be inherited, or in which order the parent classes should be inspected to find a default value. Furthermore, frames may behave as classes, as well as prototype. Last but not least, frames systems may be used define new frame systems, with, for example, other types of facets, etc.

2.6 Ontologies

An ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. According to their inventor, T. Gruber, *An ontology is a specification of a conceptualization.* For our purposes, it is a formal and declarative representation which includes the vocabulary (or names) for referring to the terms in that subject area and the logical statements that describe what the terms are, how they are related to each other, and how they can or cannot be related to each other. Ontologies therefore provide a vocabulary for representing and communicating knowledge about some topic and a set of relationships that hold among the terms in that vocabulary.

Ontological representation systems (like Ontolingua [6]) allow one to define the concepts he/she will work with. Dislike the other representation methodologies presented above, there is absolutely no inference mechanism associated with ontologies. For example, logic is associated with deduction, production rules with forward or backward chaining, etc. But ontologies remain totally at the vocabulary level.

Let's take the example of the concept of `person`. The following expressions state respectively: `person` is a concept, `birthyear` is a function, the birthyear of a person is a number, so is its age, and, finally, a relation between the age and the birthyear is established.

```
(defconcept person)
(deffunction birthyear)
(=> (person ?x)
    (number (birthyear ?x)))
(=> (person ?x)
    (number (age ?x)))
(=> (person ?x)
    (= (age ?x) (- (date) (birthyear ?x))))
```

Of course, many other such assertions are necessary to cover the

domain of husbands and wives. And, how can an interpreter do something with this is yet another problem...

Like other representation schemes, ontologies bring a number of problems: logic is not completely computable, conceptual graphs restrict the expressive power. Ontologies raise an epistemological problem on the nature of knowledge: ontological representation aims at providing means for reusing and sharing knowledge across heterogeneous systems, but, this makes the implicit assumption that knowledge would exist as such, independently on any human practice. This very Platonist approach is condemned to fail by recent research in the domain of human psychology. However this shows how a very recent technique may ultimately rely on much older mental schemes than, for example, Minksy's frames, which, 20 years ago already stressed the multiple facets of knowledge.

Nevertheless rigorous ontological foundation for knowledge representation can improve the quality of the knowledge engineering process, making it easier to build at least understandable (if not reusable) knowledge bases.

3 Conclusion

This document presented the foundations of the most fundamental representation systems.

However, we did not address a number of issues, which would have led us outside the boundaries of a simple survey, among which are the problems of uncertainty (non reliable knowledge), fuzziness (approximate knowledge), non-monotony (evolving knowledge) and other non standard logics that have sometimes to be represented in computers. We strongly advise the reader to refer to [4] for such discussions.

References

[1] A. Barr and E.A. Feigenbaum *The Handbook of Artificial Intelligence.* Volume 1. Kaufmann, 1981.

[2] M.A. Boden, editor. *Artificial Intelligence.* Academic Press, 1996.

[3] A. Bonnet. *L'intelligence artificielle, promesses et réalités.* InterEditions, 1984.

[4] R.J. Brachman and H.J. Levesque. *Readings in Knowledge representation.* Morgan Kaufmann Publishers, 1985.

[5] M. R. Genesereth and R. E. Fikes. *Knowledge Interchange Format.* Logic Group Technical Report Logic-92-1, Stanford University, 1992.

[6] T. R. Gruber. *A Translation Approach to Portable Ontology Specifications.* Knowledge Acquisition, 5(2):199-220, 1993.

[7] R. de Hoog et al. The Common KADS Model Set. U. of Amsterdam technical report. 1994.

[8] K. Van Marcke. *The Use and Implementation of the Representation Language KRS.* VUB AI Lab PhD Thesis 88-2, 1988.

[9] M. Minsky. *A Framework for Representing Knowledge*, in The Psychology of Computer Vision, P. Winston (Ed.), McGraw-Hill, 1975.

[10] A. Newell. *The knowledge level* In Artificial Intelligence, vol 18, 1982.

[11] E. Rich. *Artfificial Intelligence.* Mc Graw-Hill, 1983.

[12] S.C. Shapiro, editor. Encyclopedia of Artificial Intelligence

[13] L. Steels. Components of Expertise. In AI Magazine 11(2), 1990.

[14] S.L. Tanimoto. *The elements of Artificial Intelligence.* Computer Science Press, 1987.

[15] B. Wielinga. The KADS Knowledge Modeling Approach. In Proceedings of the 2nd Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop. 1992.

2. Integrated Model for inner Niger Delta in Mali

2.1 Model description

This section describes the integrated model for the inner Niger Delta in Mali that has been realised in the SIMES context.

Thème : Gestion des Ressources en Eau

MODELISATION INTEGREE D'UN ECOSYSTEME INONDE ET GESTION DES EAUX : LE CAS DU DELTA INTERIEUR DU NIGER AU MALI

Marcel KUPER[1], Christian MULLON[2],
Yveline PONCET[2], Didier ORANGE [1]et Pierre MORAND [1]

[1] *IRD (ex-ORSTOM), BP 84, Bamako, MALI*
[2] *ERMES, IRD, Université Orléans, FRANCE*

Abstract

An integrated model of the Niger river inland delta is presented. The model represents the hydrological functioning of the delta, as well as the natural resources which are largely dependant on the availability of water. The management and exploitation of these resources (agriculture, fisheries, livestock) is also covered by the model. The model is based on different thematic research studies and was developed while these thematic studies were on-going. This permitted the use of the model as a platform for discussion between researchers of different disciplines, but also allowed us to engage discussions with policy makers, line agencies and experts on on-going research. The model is used to carry out simulations and display graphic animations, allowing a better understanding of the functioning of the delta. The model is also used to carry out sensitivity analyses for different parameters such as the availability of water and the fertility of certain areas. Finally, the model is used to determine the values of parameters that influence the production of the different exploitation systems.

INTRODUCTION

Le Delta Intérieur du Fleuve Niger au Mali est un exemple d'écosystème où régime hydrologique, dynamique de l'environnement et de la faune aquatique et activités humaines (pêche, agriculture, élevage) sont étroitement associés. La durabilité des modes d'exploitation par l'homme d'un tel milieu est une question fondamentale, ce système se trouvant en position intermédiaire entre des systèmes quasi-naturels (Bassin de l'Amazone) et des systèmes fluviaux fortement modifiés par l'homme (aménagements sur le Sénégal). Un enjeu majeur du développement est de passer de l'exploitation à la gestion des ressources naturelles en tenant compte (1) de la dynamique à long terme sur les systèmes physiques, biologiques et anthropiques et (2) des besoins et des usages des populations.

Les écosystèmes sont complexes, entraînant des critères et contraintes de décision multiples. C'est pourquoi le fait de disposer de nombreuses informations actualisées et synthétiques ne suffit pas forcément pour mesurer les implications à venir, à moyen ou long-terme, d'une décision portant par exemple sur un nouvel aménagement physique ou sur un changement de réglementation, sur laquelle il sera difficile de revenir. Ce type d'outils destiné à explorer et à discuter les stratégies de gestion possibles et les scénarios résultants est donc indispensable : il relève de la modélisation par simulation. Les phases de conception et d'expérimentation de ce genre d'outils sont du ressort de la recherche, du fait que la part de l'effort d'innovation est alors très importante, alors que les phases d'implémentation, d'utilisation et d'extension des dispositifs concernent davantage les bureaux d'études et services techniques nationaux, et sont alors du ressort d'activités de développement.

Dans un contexte de développement durable, le premier objectif d'une modélisation d'un système complexe est de représenter des relations spatiales et temporelles entre les différents niveaux d'organisation et donc de formaliser des emboîtements d'échelles multiples. C'est en outre de permettre d'anticiper les dynamiques des systèmes étudiés, afin, par exemple, de contrôler les impacts d'une stratégie de gestion. C'est enfin, en dernière étape, de permettre l'implication des différents acteurs et en particulier la conviction des décideurs ; le modèle doit donc pouvoir représenter la dynamique du système.

L'objectif du projet GIHREX[1] de l'IRD est ainsi la compréhension de la dynamique naturelle du Delta, la connaissance des modes d'organisation et d'exploitation (sociale, technique et économique) par l'homme et l'analyse de leur durabilité. La modélisation est, dans le programme GIHREX utilisée comme un outil permettant de combiner une approche écologique de la dynamique des ressources et une approche socio-économique des activités humaines, de formaliser des interactions spatiales dans une dynamique temporelle, et ainsi de mieux comprendre le fonctionnement d'un écosystème exploité. Donc la modélisation participe à trois objectifs scientifiques forts :

1. la description de l'objet d'étude,
2. l'analyse et la compréhension du fonctionnement de l'objet,
3. la simulation et l'évolution de l'objet.

Concept de modélisation intégrée

La modélisation de systèmes complexes est un domaine de recherche particulièrement actif actuellement, impliquant mathématique, informatique, statistique ; mais toutes les avancées dans ce domaine ont été basées sur des développements dans des disciplines appliquées. Des outils mathématiques, informatiques, statistiques, des architectures informatiques appropriées sont développés qui permettent d'envisager de pouvoir bientôt répondre aux objectifs de développement exposés ci-dessus. Or, il s'avère que la question de la modélisation des écosystèmes exploités, compte tenu de la diversité des processus à représenter, de la multiplicité des échelles et des niveaux d'organisation, demande un effort de recherche spécifique ; il faut plus qu'utiliser ou adapter les outils existants ; il faut en développer des spécifiques ; il faut conjointement réfléchir à une méthodologie de leur emploi. Le questionnement principal est le suivant : Comment formaliser les systèmes physiques, biologiques et anthropiques de manière à anticiper leur dynamique ? Ce questionnement scientifique passe par cinq étapes de réflexion structurant notre démarche de recherche (Kuper, 1997 ; Strosser, 1997):

1. définir les sous-sytèmes en interaction
2. définir les relations d'échelle temps / espace, inter et intra (e.g. Blöschl et Sivapalan, 1995)
3. choix de construction pour chaque sous-système
4. tester la stabilité numérique de la somme des constructions
5. calage, validation, robustesse

La réalisation de ces cinq étapes permettra de répondre aux deux sous-questions suivantes :

1. *Peut-on représenter (formaliser, modéliser) un tel système ?*
2. *Avec quelle validité ?*

Il se trouve que la modélisation de l'écosystème exploité du Delta Intérieur du Niger, constitue un cas d'étude exemplaire, permettant de contribuer à l'élaboration des outils spécifiques et à l'émergence d'une méthodologie adaptée. Cela s'est concrétisé dans la définition des processus à retenir, des niveaux d'organisation à considérer, des scénarios de modélisation à mettre en oeuvre. Cela est du à notre avis :

- A la bonne connaissance antérieure de ce système (e.g. Brunet-Moret et al., 1986 ; Olivry, 1993 ; Quensière, 1994 ;)

[1] Gestion Intégrée, Hydrologie, Ressources et Systèmes d'EXploitation.

- A l'échelle large retenue : celle de tout le Delta,
- A la perspective de gestion,
- Aux premiers choix en matière d'architecture informatique , basée sur l'imbrication de couches correspondant aux niveaux d'organisation.

On considère donc le modèle intégré comme un outil qui nourrit les négociations des acteurs, en quantifiant l'impact des événements ou interventions sur le fonctionnement du delta, par exemple l'effet du développement des petits périmètres irrigués sur le fonctionnement hydrologique du delta (écoulements, surface inondée). Ainsi, le modèle devient un élément d'une approche ou analyse intégrée dont le caractère est itératif : le modèle intégré va nous permettre de mieux expliciter ou détailler les questions pertinentes de gestion, en utilisant ce modèle comme plate-forme de discussion (cf. Dzeakou et al., 1998). Ceci explique l'intérêt de développer dès le début du projet une maquette du modèle intégré, afin d'entamer cette discussion et mieux répondre aux attentes des acteurs. Ceci a permis d'une part, un développement progressif de la maquette, mais surtout une prise de conscience par les acteurs de la nécessité d'intégration des actions de développement à mener. La maquette de modélisation intégrée tient donc une place centrale dans la circulation de l'information (fig.1).

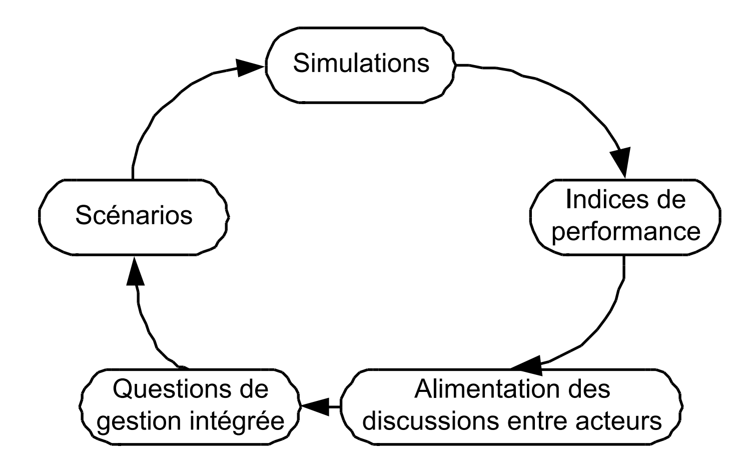


Figure 1 : Rôle moteur de circulation de l'information d'une maquette de modélisation intégrée

Le développement et la mise en œuvre d'une modélisation intégrée joue également un rôle fédérateur et stimulateur des différentes disciplines scientifiques allant des sciences de la nature aux sciences humaines.

Pour construire leurs modèles, les modélisateurs exigent des informations au fur et à mesure de leurs avancées. Ils expriment alors leurs besoins aux chercheurs thématiques, ce qui a pour conséquence d'orienter leurs recherches. Les modélisateurs peuvent, également à leur tour, apporter leurs connaissances aux thématiciens : par exemple, en comparant les résultats obtenus après simulation avec les données observées dans la réalité, ils peuvent tirer des conclusions qui sont autant d'éclaircissements supplémentaires.

Principes de la maquette MIDIN

Modélisation intégrée d'un écosystème inondable

Le projet est organisé autour du développement d'une maquette de modélisation intégrée du Delta Intérieur du Niger et concerne plus spécifiquement les sciences suivantes : hydrologie, hydrochimie, hydrobiologie, sociologie, géographie, agronomie et sciences de l'information.

L'eau est l'élément intégrateur de toutes les disciplines environnementales concernées dans cette étude sur la dynamique spatio-temporelle des ressources naturelles du delta intérieur du Niger. Sa disponibilité est à l'origine de tous les questionnements, du milieu physique à l'homme, à savoir :

- Quelle est la variabilité des ressources en eau dans l'espace et dans le temps ?
- Comment cette dynamique spatio-temporelle de la disponibilité en eau influence la dynamique physique du milieu (eaux et flux géochimiques), génère et entretient la biodiversité et la productivité des ressources naturelles de cet écosystème ?
- Quelle est son influence sur les déterminants socio-économiques qui contrôlent la pression sur les ressources naturelles et leur mode d'exploitation ?

Ainsi, la variabilité (dans le temps et dans l'espace) de la disponibilité en eau, apparaît être l'élément majeur d'explication du comportement de tous les autres paramètres du système. Cette disponibilité en eau dépend à la fois de la variabilité climatique, de la morphologie du milieu et des aménagements (fig. 2).

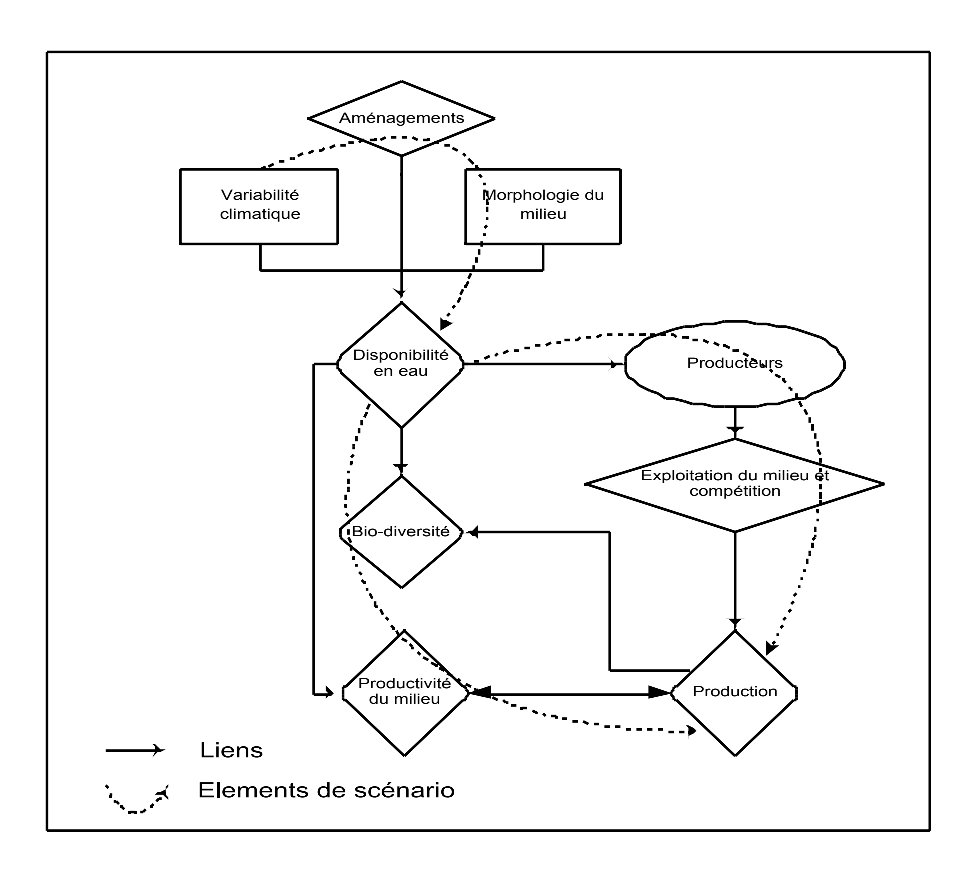


Figure 2 : Cheminement des réflexions thématiques de la variabilité climatique aux productions du système

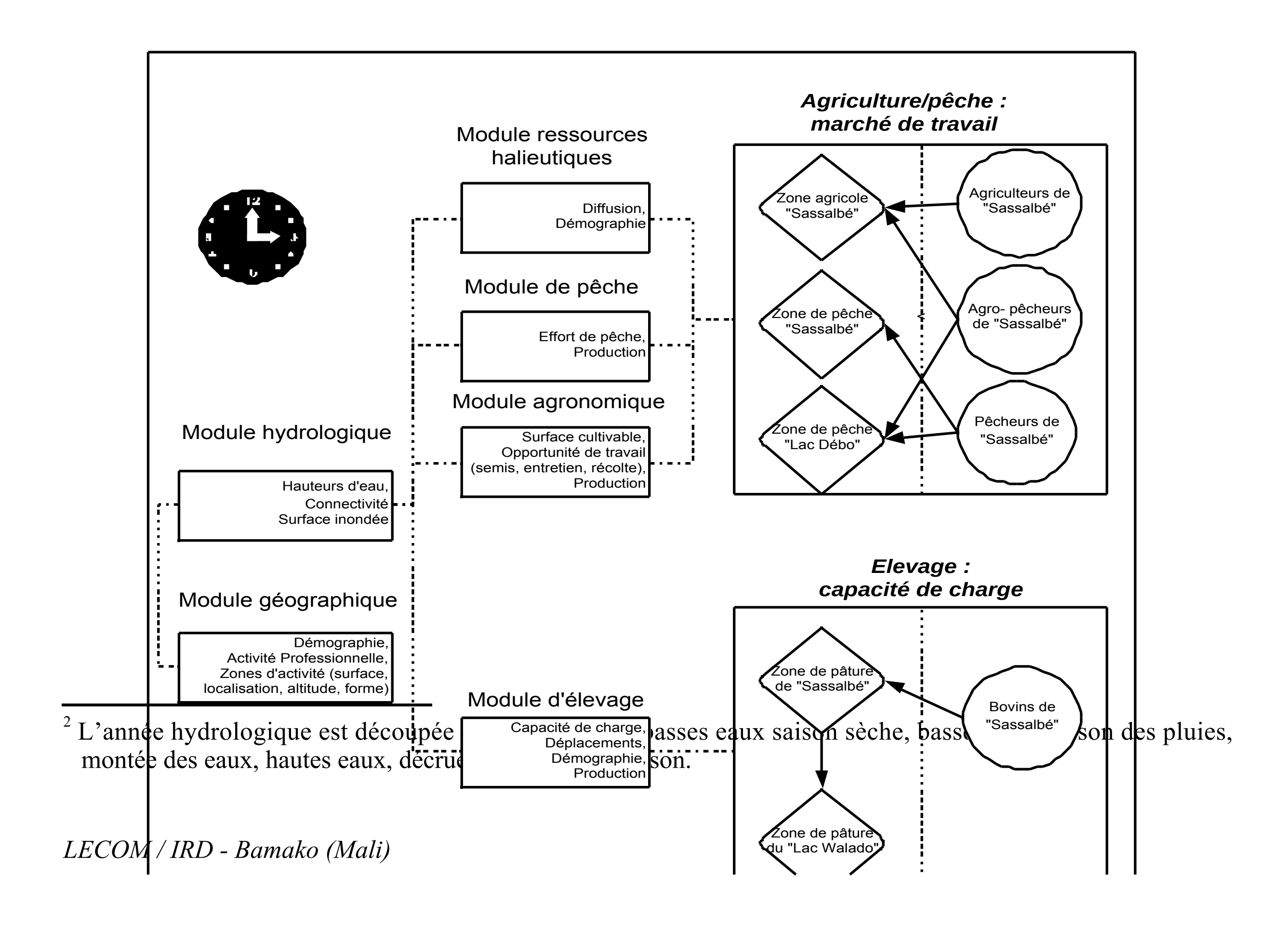
En fonction de cette disponibilité en eau, on définit les productions possibles en ressources naturelles du système en suivant parallèlement deux cheminements scientifiques différents mais convergents : un cheminement via les sciences physiques, un cheminement via les sciences technotopes (fig. 2). Le premier passe par la compréhension des processus de mobilisation des matières (chemins de l'eau et érosion, éléments chimiques utiles à l'établissement des chaînes trophiques) puis des processus liés à la biodiversité et la productivité du milieu. Le second dépend des choix et stratégies des producteurs et passe donc par la compréhension de la dynamique de l'exploitation du milieu et la compétition pour les ressources naturelles. On note que les aménagements d'une part, et les choix stratégiques d'autre part, sont des termes sur lesquels la modélisation pourra construire des scénarios.

Le cas du delta intérieur du Niger

La configuration de la maquette MIDIN (Modélisation Intégrée du Delta Intérieur du Niger) est modulaire, avec des modules thématiques. L'avantage d'une telle configuration est d'une part, qu'on peut développer et améliorer ces modules indépendamment, et d'autre part, qu'on peut assurer des sorties thématiques, qui peuvent être vérifiées et analysées. L'ensemble est commandé par une horloge selon l'architecture décrite en figure 3. Les modules thématiques sont :

- *la disponibilité en eau* : détermination des surfaces inondées à chaque pas de temps et pour chaque objet hydrologique (plaine, bief, nœud) à partir des hauteurs d'eau observés dans 18 stations hydrométriques (Marieu et al., 1998); un module climatique est en projet, pour caractériser des paramètres climatiques (pluie, température de l'air, nébulosité, ET_R) dans leurs dynamiques spatio-temporelle,
- *les ressources halieutiques* :ce module est basé sur le travail de Bousquet et Morand (1994) et représente la dynamique de (ré)génération et de diffusion de la quantité de poisson dans le temps[2] et dans l'espace,
- *la chaîne trophique* : ce module est en projet avec comme objectif la compréhension des processus d'édification des ressources aquatiques, les sources de la productivité et de la biodiversité, le rôle de la qualité de l'eau en intégrant sa dynamique spatio-temporelle,
- *les systèmes de production* (pêche, riziculture, élevage) : les trois filières principales de production,
- *l'exploitation et la compétition* : analyse des stratégies d'exploitation en fonction de paramètres socio-écologiques, paramétrisation des choix.

La maquette de modélisation intégrée du delta intérieur du Niger est construite à partir d'une représentation géoréférencée d'une architecture spatiale hydrologique fondée sur la structure en réseau (nœuds et flux) de l'hydrosystème et sur les fonctionnalités hydrologiques des objets géographiques (transfert, stockage, vidange). L'hydrosystème est donc représenté par des traits (chenaux, rivières, fleuve), des cônes (lacs, plaines) et des nœuds (défluences, confluences).
Les attributs des objets spatiaux concernent l'eau, moteur du système (remplissage, vidange et évaporation), mais aussi certains champs de l'écologie végétale et planctonique, les différents biotopes d'intérêt halieutique, agricole et pastorale, les lieux de résidence des groupes humains et leurs stratégies de migration pour l'exploitation des ressources naturelles. Si donc la variable explicative principale est la quantité d'eau, les variables de sorties sont les productions possibles (fig. 3) des zones de pêche, des zones agricoles et des zones pastorales.



[2] L'année hydrologique est découpée […] basses eaux saison sèche, basse[…] saison des pluies, montée des eaux, hautes eaux, décru[…]son.

Figure 3 : Configuration modulaire de la maquette de modélisation intégrée du delta intérieur du Niger (MIDIN).

La maquette est organisé autour des groupes d'exploitants[3] (pêcheurs, agriculteurs,…) qui sont rattachées à leur zone d'origine, mais qui peuvent également aller dans d'autres zones d'exploitation (*principe de mobilité*). Cette exploitation est déterminée dans le delta par des conventions traditionnelles (cf. Baumann et al., 1994, par exemple des liens de parenté (*principe de connectivité*). La modélisation obtenue correspond à « *un modèle de comportements »* et non de stratégies. En effet, la dynamique spatiale des groupes est liée à l'allocation des ressources naturelles dans les zones exploitables et non à la stratégie des choix des acteurs. Les groupes d'exploitants (plus de 200 pour l'ensemble du delta) ainsi que les zones d'exploitation (une centaine) sont décrit d'une façon explicite :

- Groupes d'exploitants : activité principale, effectif, localisation, connectivité, taux de migration,
- Zones d'exploitation : superficie, fertilité, altitude, forme

Ensuite, il s'agit de :

1. *Quantifier l'impact de la crue (« le moteur du delta ») sur la génération de la ressource,* à partir du remplissage en eau des objets hydrologiques du modèle hydrographique conceptuel du delta, basé sur une lecture des cotes limnimétriques observées sur 3 années tests (1993, 1994, 1995) et intégrant une valeur moyenne d'évaporation[4]. La productivité des milieux est modélisées à partir de relations simples *volume d'eau / abondance de la ressource halieutique[5], agronomique (superficie cultivable) et pastorale* ;
2. *Représenter les mouvements de population pour l'exploitation des ressources naturelles,* on utilise des modèles de type « *marché de l'emploi* » en distinguant les activités halieutiques et agricoles représentées par un modèle *pulling* et les activités pastorales représentées par un modèle *pushing*.

    les activités halieutiques et agricoles : un modèle pulling
    Les populations du delta sont représentées par des groupes (groupes d'agriculteurs, groupes d'agro-pêcheurs, groupes de pêcheurs) qui migrent vers des zones (zones de pêche, zones agricoles) lorsque les ressources exploitables de ces zones sont suffisantes : « les zones attirent les groupes », notion d'abondance (pêche) et de fertilité (agriculture) des zones. Dans la figure 2, nous avons pris l'exemple de la plaine de « Sassalbé », une zone d'environ 700 km_ avec forte présence d'agriculteurs, d'éleveurs et pêcheurs. Après la récolte de riz en décembre, les agro-pêcheurs de cette zone se déplacent traditionnellement vers d'autres zones de pêche telle que le lac Débo.

    les activités pastorales : un modèle pushing
    Les populations d'éleveurs ne sont pas représentées. On a représenté les groupes de bétail (de troupeaux), qui se déplacent quand il n'y a plus d'herbe dans la zone pastorale où se trouve le troupeau : « les zones chassent les groupes », notion de capacité d'accueil des zones. Ainsi, les troupeaux de Sassalbé se déplacent vers le nord

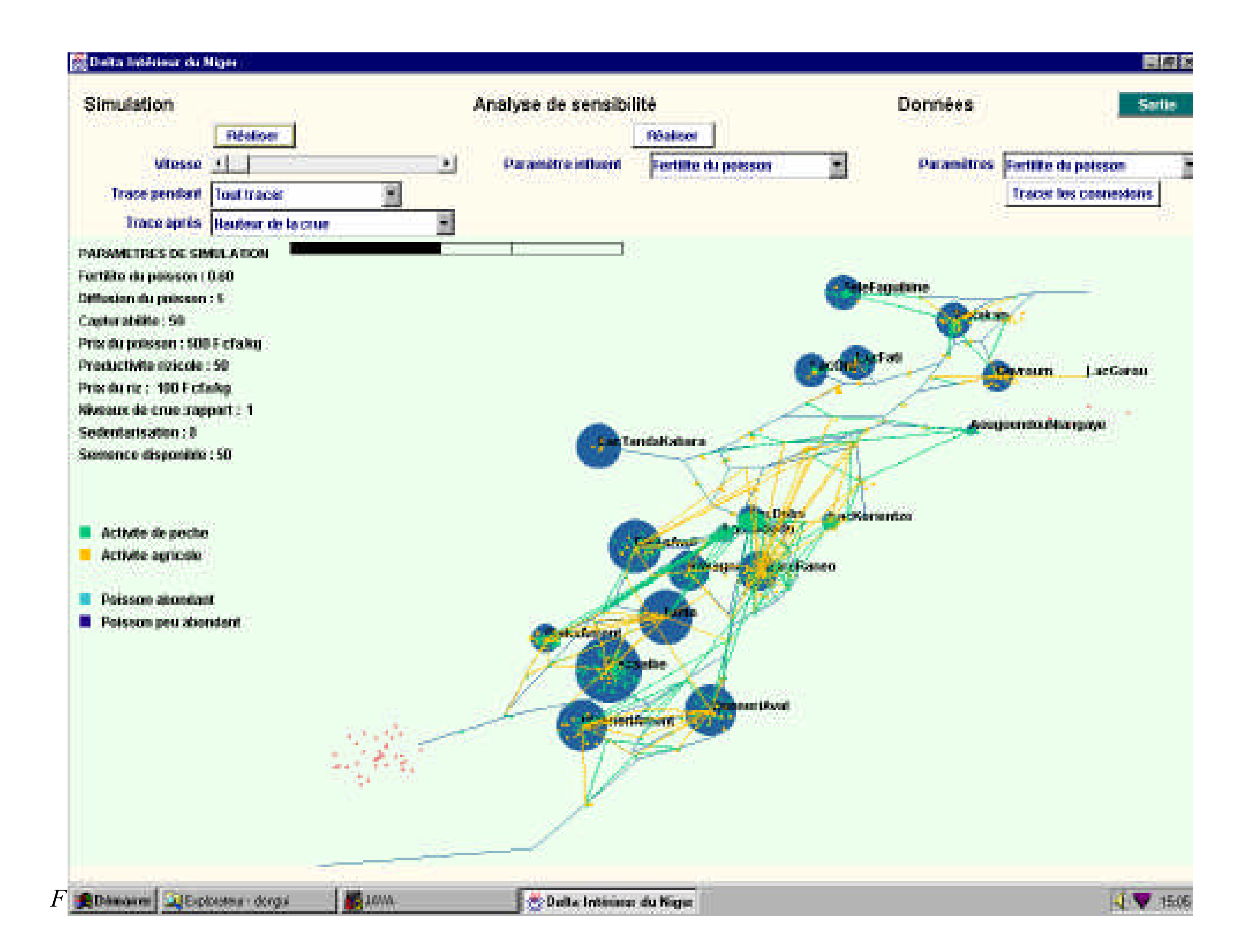
[3] Dans le delta, il existe environ 200 000 ménages, qui participent souvent à plusieurs activités professionnelles (pêche, agriculture, élevage). Ces ménages prennent des décisions concernant le choix d'activité (agriculture, pêche, élevage ou autre), la durée de l'activité, le choix de sites (parcelles, sites de pêche, trajet dans l'espace et dans le temps pour l'élevage), et l'utilisation de leur effectif.
[4] La mise au point d'un modèle hydraulique est en cours de réalisation à partir du logiciel SIC (Cemagref, 1992), il prévoit la connaissance du débit, hauteur et vitesse d'eau aux nœuds du modèle hydrographique conceptuel en fonction du débit dans le haut bassin du Niger/Bani
[5] Selon les conclusions du projet de recherche *Pêche dans le Delta Central du Niger* (Quensière, 1994)

vers des zones de pâture autour du complexe lacustre (lacs Walado, Débo) en saison sèche.

Présentation de la maquette MIDIN

L'interface de la maquette MIDIN est présentée en figure 4. Dans l'état actuel de la conception, la maquette a trois fonctions différentes :

- Simulation et animation graphique de l'ensemble des processus modélisés (figure 4). Il est possible d'animer un seul processus,
- Analyses de sensibilité en faisant varier certains paramètres influents, par exemple de 0 à 120 % par tranche de 10 %,
- Exécution de scénarios en accordant des valeurs aux paramètres de simulation (fertilité du poisson, etc.).



F

Modélisation intégrée et systèmes d'observation

La maquette de modélisation intégrée participe à un plus grand ensemble qu'est le système d'information environnementale, élément structurant d'un observatoire environnemental du delta intérieur du Niger. Dans l'articulation préconisée des différentes activités de recherche, un mécanisme itératif (ou "feed-back") permet de réajuster, en permanence, les calibrations prises dans chaque composante, en fonction des besoins rencontrés dans les autres composantes : ***aller-retour permanents entre les données issues des systèmes d'observation, les résultats des simulations de la modélisation intégrée et les connaissances des processus acquises par les recherches thématiques*** (fig. 5).

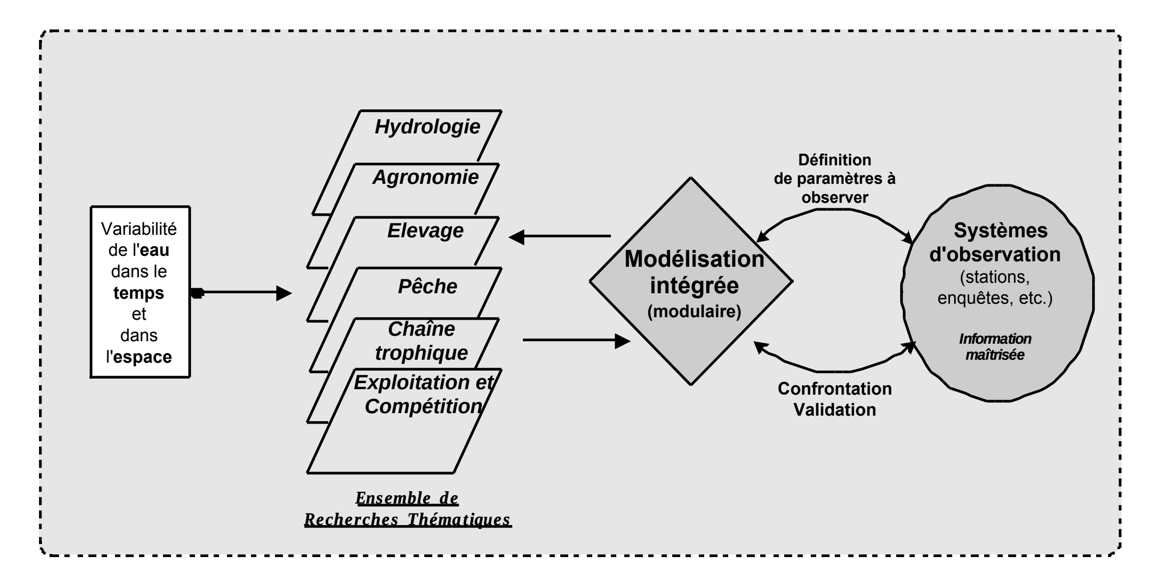


Figure 5 : Articulation entre les recherches thématiques, la modélisation intégrée et les systèmes d'observation au sein d'une structure observatoire

CONCLUSION

Le premier objectif d'une modélisation d'un système complexe est de représenter des relations spatiales et temporelles entre les différents niveaux d'organisation et donc de formaliser des emboîtements d'échelles multiples. Cette formalisation des liens entre les différents compartiments de l'écosystème doit permettre d'anticiper les dynamiques des systèmes étudiés afin d'une part, de vérifier (ex-ante) les impacts d'une stratégie de gestion et d'autre part, de favoriser l'implication des différents acteurs, et en particulier la conviction des décideurs.

Dans un contexte de développement durable, le modèle s'attachera surtout à représenter la dynamique du système en fonction des systèmes de production. Dans le cas du delta intérieur du Niger au Mali, les développeurs, aménageurs, décideurs, producteurs ont une question récurrente, à savoir : quel est l'impact des infrastructures hydrauliques (barrage, aménagements hydro-agricoles tel l'Office du Niger,…) sur la production des ressources naturelles et la gestion des systèmes d'exploitation ?

La maquette du modèle intégré construite pour le delta intérieur du Niger peut servir à faire un diagnostic de la situation actuelle pour des questions précises de gestion des eaux et évaluer l'impact de différents événements ou interventions sur le fonctionnement du delta. Dans notre cas, un intérêt particulier est apporté au fonctionnement hydrologique en relation avec les productions agricole/halieutique/pastorale.

Enfin, ces recherches thématiques ont participé à la constitution d'une base de connaissance venant nourrir la maquette du modèle intégré pour la réalisation de ses deux fonctions principales :

1. la *simulation*, pour reproduire artificiellement un phénomène en fixant d'autres paramètres ; par exemple, simuler l'implantation d'un barrage et évaluer son impact sur le fleuve Niger et sur la production de riz ;
2. la *prévision*, pour envisager en avance l'évolution d'un milieu et son exploitation en fonction de certaines hypothèses ; par exemple, prévoir la hauteur d'eau à Mopti en fonction des pluies du haut bassin du Niger et du Bani et évaluer leur impact sur la production de poissons.

Le développement d'un tel modèle intégré du delta intérieur du Niger est donc bien fédérateur et stimulateur des différentes disciplines scientifiques. De plus, il aboutit à une représentation du fonctionnement d'un écosystème complexe en fonction de son utilisation par l'homme et devrait répondre aux questions des acteurs désirant le gérer durablement.

Références

Baumann E., Fay C., et Kassibo B. 1994. *Systèmes de production et d'activité : trois études régionales*. In : Quensière J. (ed.). La pêche dans le delta central du Niger.

Blöschl G. et M. Sivapalan. 1995. Scale issues in hydrological modeling : a review. *J. of Hydrological Processes, Vol. 9, Issues 3-4*.

Bousquet F et P. Morand. 1994. *Modélisation de la ressource : relations entre l'effort de pêche, la dynamique du peuplement ichtyologique et le niveau des captures dans un système fleuve-plaine*. In : Quensière J. (ed.). 1994. *La pêche dans le delta central du Niger.* Volume 1 et 2. IER-Orstom-Karthala. Paris.

Brunet-Moret Y., Chaperon P., Lamagat J.P., Molinier M. 1986. *Monographie du Niger*. Deux tomes. Orstom, Paris.

Cemagref. 1992. SIC, Simulation of Irrigation Canals : User's Guide and Theoretical Concepts. *Cemagref, Montpellier, France*.

Dzeakou P., M. Kuper, P. Morand, C. Mullon, Y. Poncet. 1998. *Modélisation Intégrée du Delta Intérieur du Niger: Développement d'une Maquette du Modèle Intégré*. IRD, Bamako, Mali.

Kuper M. 1997. *Irrigation management strategies for improved salinity and sodicity control*. Thèse de doctorat, Université Agronomique de Wageningen, Pays-Bas.

Marieu B., F. Bamba, J. Briquet, N. Cissé, M. Gréard, T. Henry des Tureaux, G. Mahé, A. Mahieux, J.C. Olivry, D. Orange, C. Picouet, M. Sidibé, M. Touré. 1998. *Actualisation des données hydrométriques du fleuve Niger au Mali pour EQUANIS*. ORSTOM/DNHE, Bamako, Mali.

Olivry J.C. 1993. *Fonctionnement hydrologique de la cuvette lacustre du Niger et essai de modélisation de l'inondation du delta intérieur.* In : Grands bassins fluviaux périatlantiques : Congo, Niger, Amazone, publié par Olivry J.C., Boulègue J. Orstom éditions.

Quensière J. (ed.). 1994. *La pêche dans le delta central du Niger.* Volume 1 et 2. IER-Orstom-Karthala. Paris.

Strosser P. 1997. *Analyzing alternative policy instruments for the irrigation sector.* Thèse de doctorat, Université Agronomique de Wageningen, Pays-Bas.

2.2 User manual

This section is the user manual for the prototype of the integrated model.

MODELISATION INTEGREE DU DELTA INTERIEUR DU NIGER

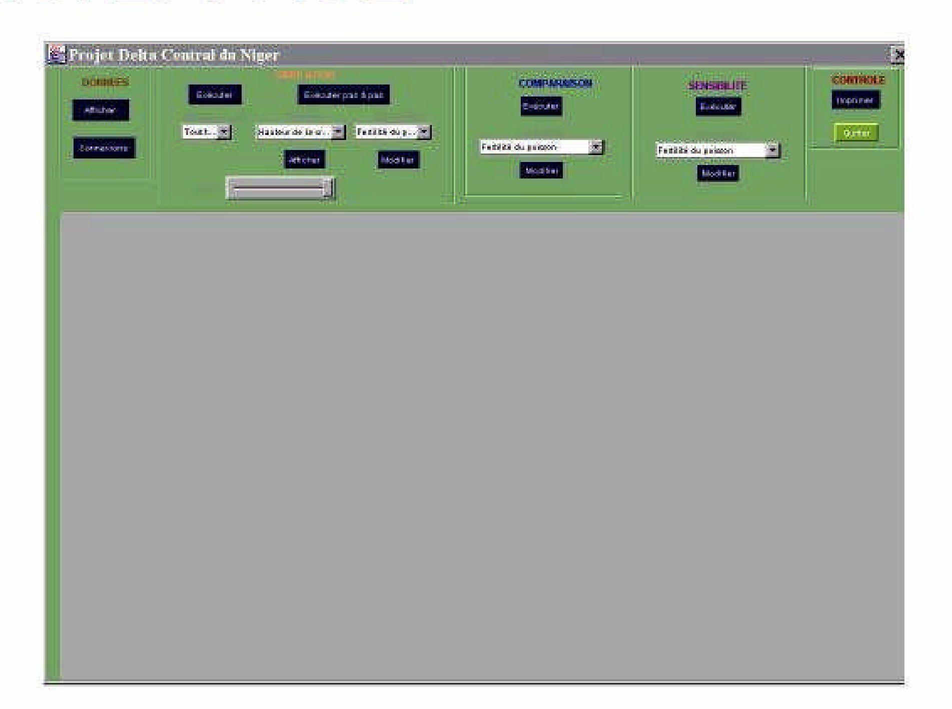
IRD

M. Kuper, Y. Poncet, C. Mullon, P. Morand, E. Benga

Manuel d'utilisation

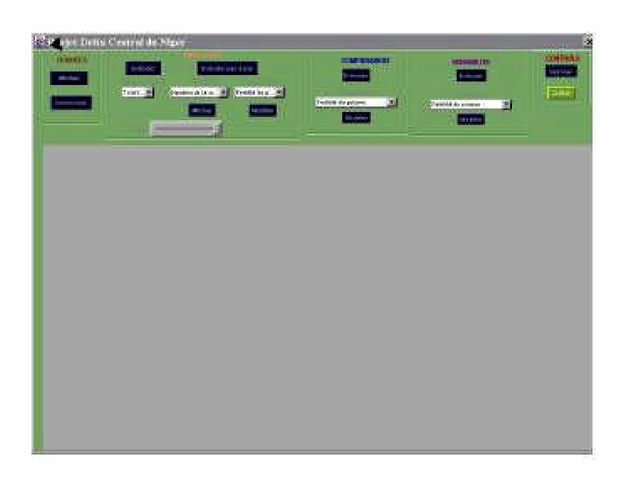
MANUEL D'UTILISATION

INTERFACE



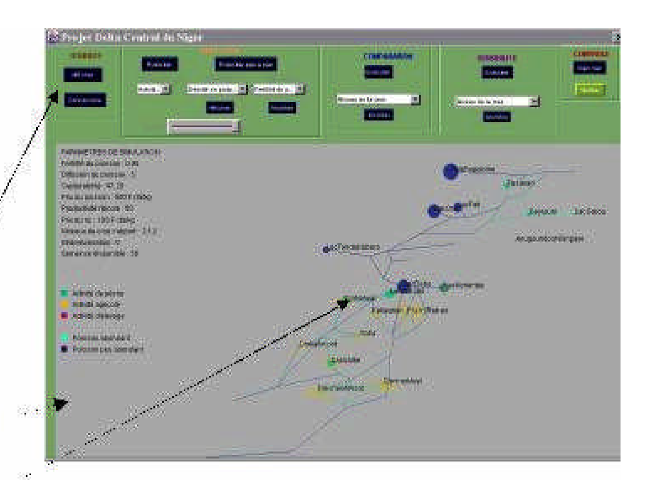
DONNEES

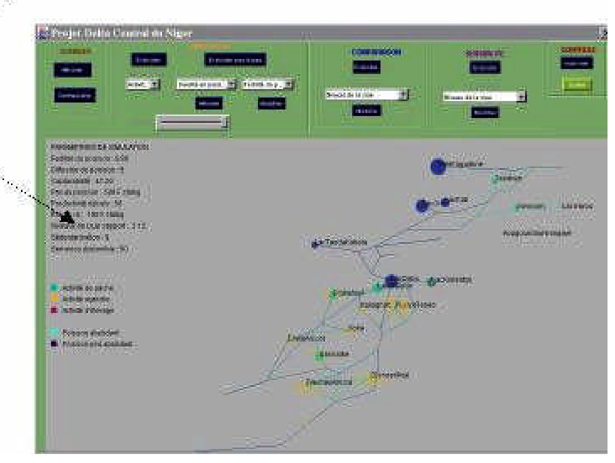
- Afficher
- Connexions



DONNEES

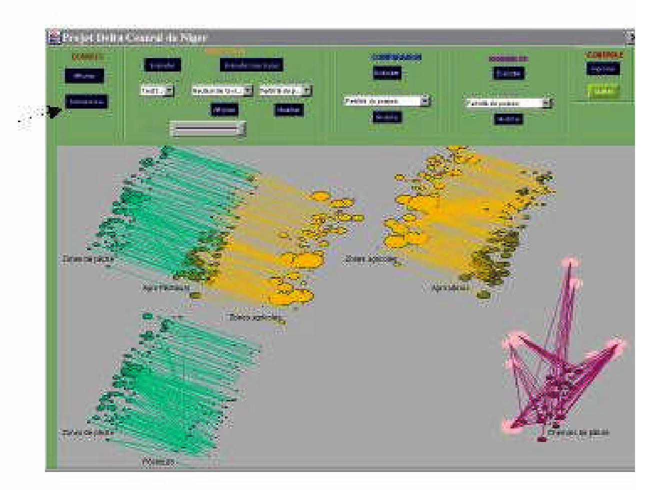
- Afficher les données
  – clic sur afficher
  – une carte de la zone s 'affiche
  – clic sur un des lieux
  – les données correspondantes s 'affichent
  – clic n 'importe où pour réafficher la carte de la zone





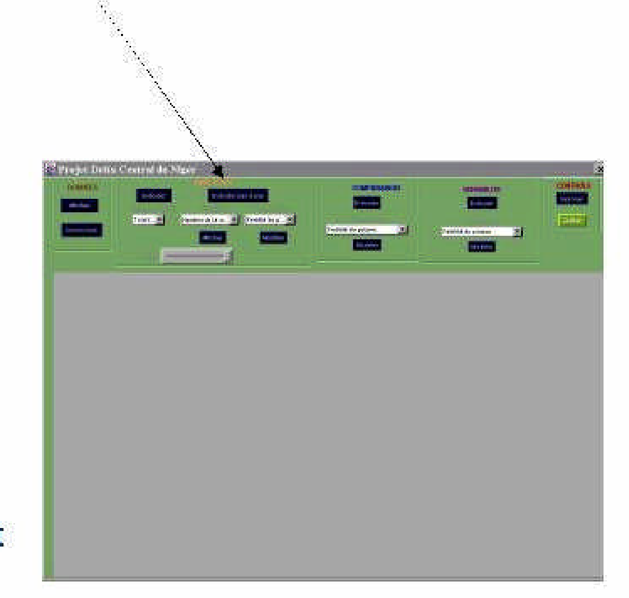
DONNEES

- Connexions
  – Clic sur connexions



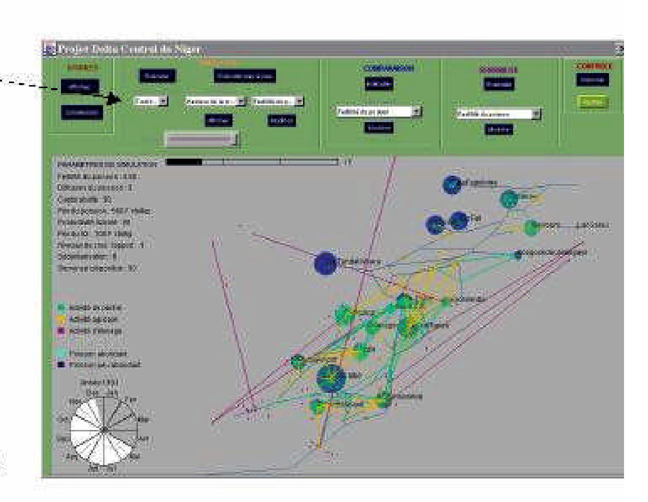
SIMULATION

- Choisir les options de tracé
- Modifier la valeur d'un paramètre
- Changer la vitesse de simulation
- Simuler
- Simuler pas à pas
- Choisir les résultats à représenter
- Représenter localement les résultats



SIMULATION

- Choisir les options de tracé :
  - Dérouler le menu
  - Déplacer le curseur jusqu'à l'option choisie

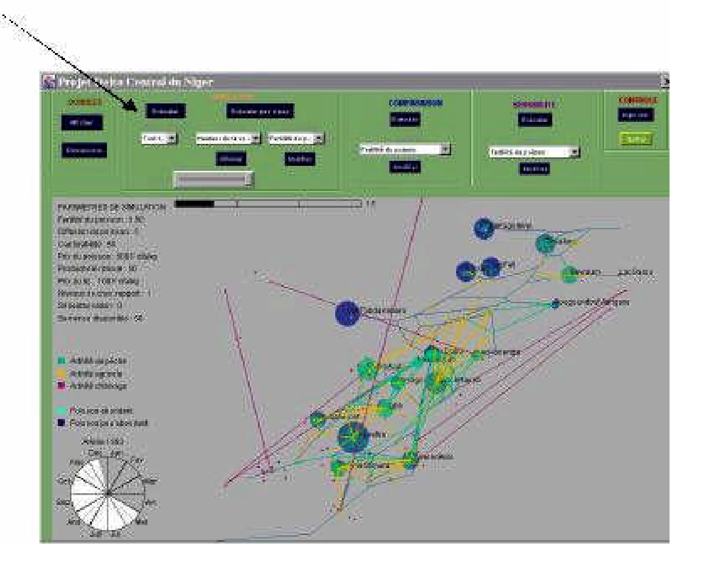


Les options de tracé sont :

tout tracer
ne tracer que la crue
tracer l'activité agricole
tracer l'activité de pêche
tracer l'activité d 'élevage

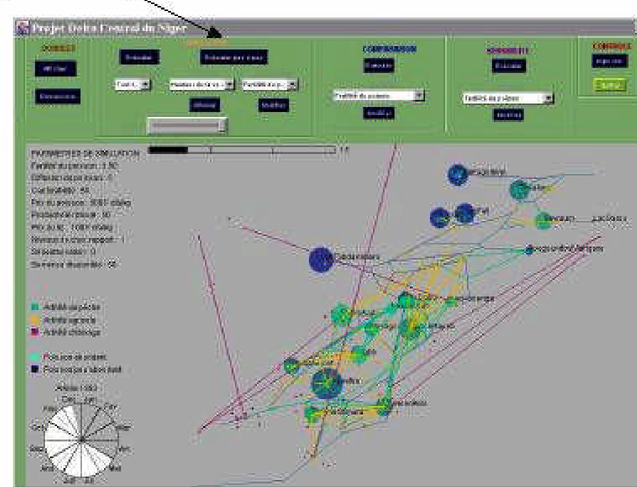
SIMULATION

- Simuler :
  - Clic sur Exécuter



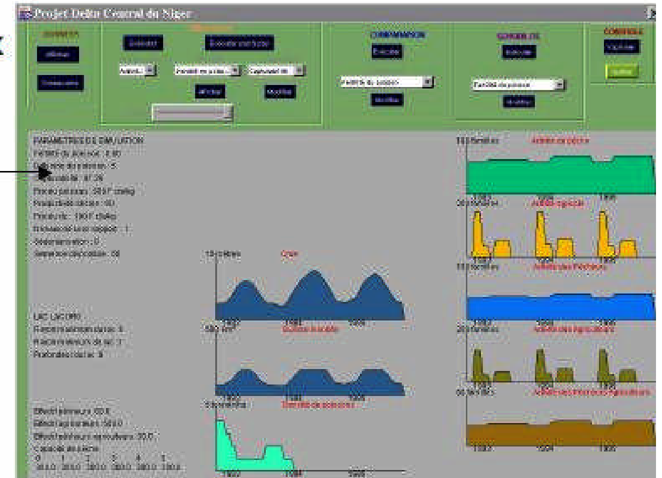
SIMULATION

- Simuler pas à pas :
  - Clic sur Exécuter pas à pas
  - Clic n 'importe où pour avancer d 'un pas



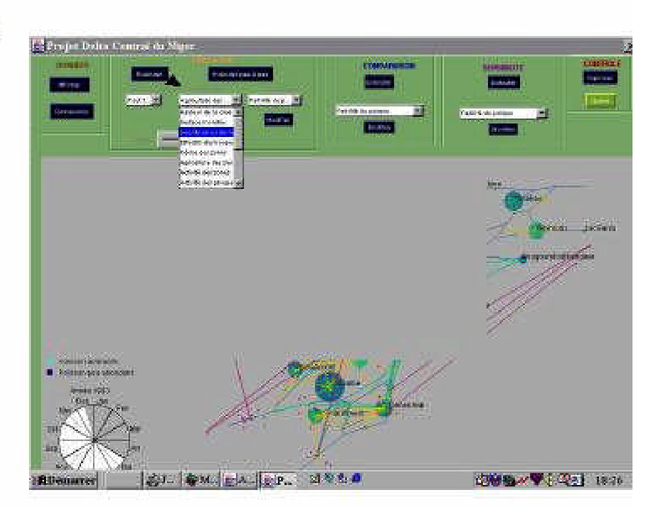
SIMULATION

- Représenter localement les résultats :
  - à la fin d 'une simulation
  - clic sur un des lieux
  - les données correspondantes s 'affichent
  - clic n 'importe où pour réafficher la carte de la zone



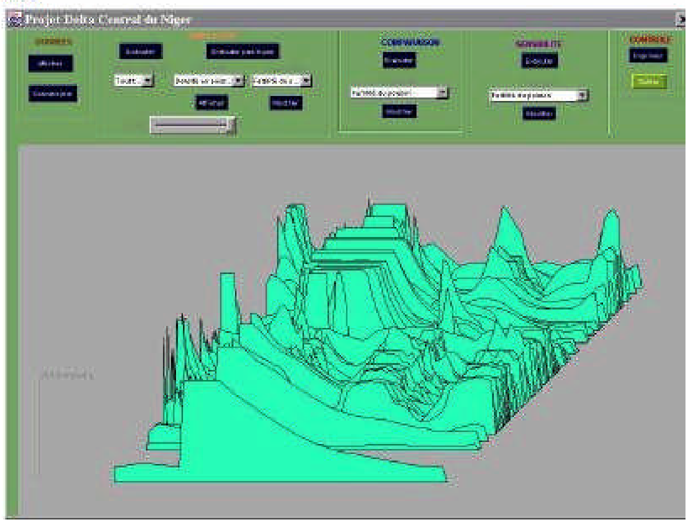
SIMULATION

- Afficher les résultats :
  - Dérouler le menu
  - Déplacer le curseur jusqu'à l'option choisie
  - Relâcher le curseur
  - Clic sur le bouton Afficher



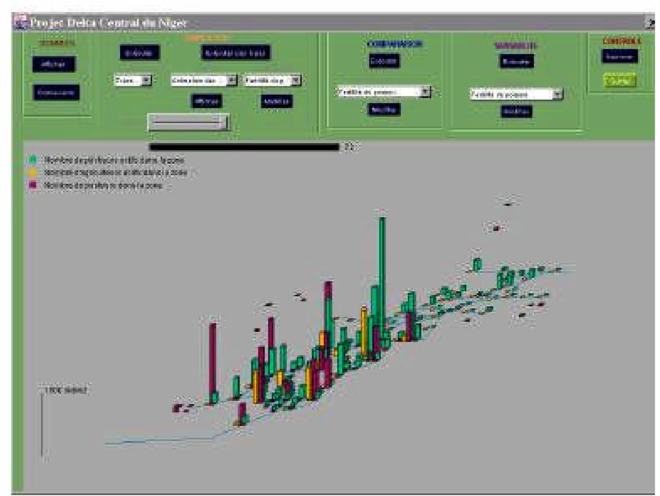
SIMULATION

- Afficher les résultats :
  - Exemple : répartition de la densité en poissons



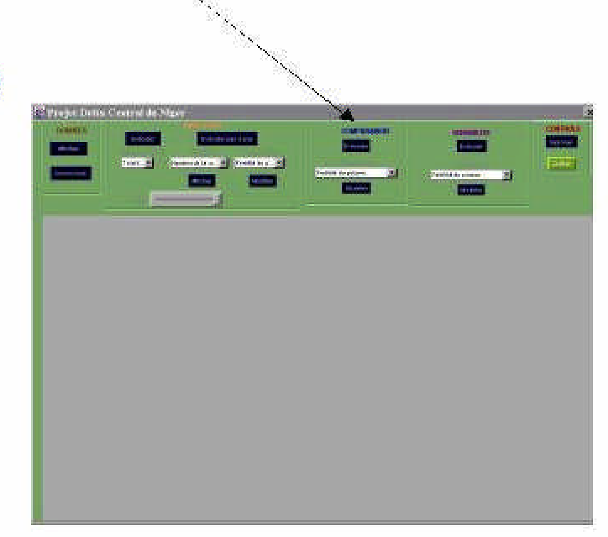
SIMULATION

- Afficher les résultats :
  - Exemple : activités dans le delta (graphique animé)



COMPARAISON

- Choisir le paramètre de comparaison
- Fixer les valeurs du paramètre de comparaison
- Effectuer

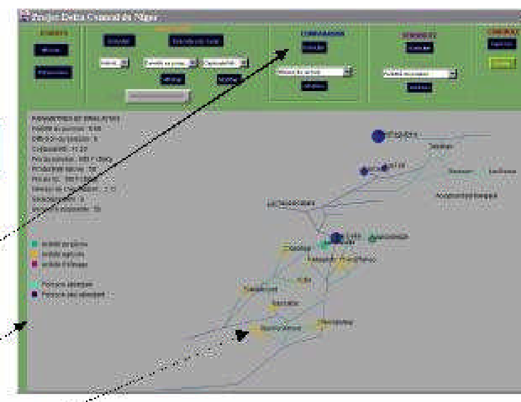


COMPARAISON

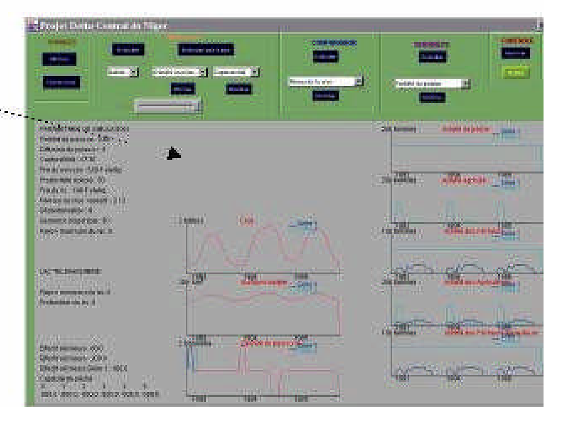
- Choisir le paramètre de comparaison
- Fixer les valeurs du paramètre de comparaison



COMPARAISON



- Effectuer :
  - clic sur Exécuter
  - une carte de la zone s 'affiche
  - clic sur un des lieux
  - les comparaisons correspondantes s 'affichent
  - clic n 'importe où pour réafficher la carte de la zone

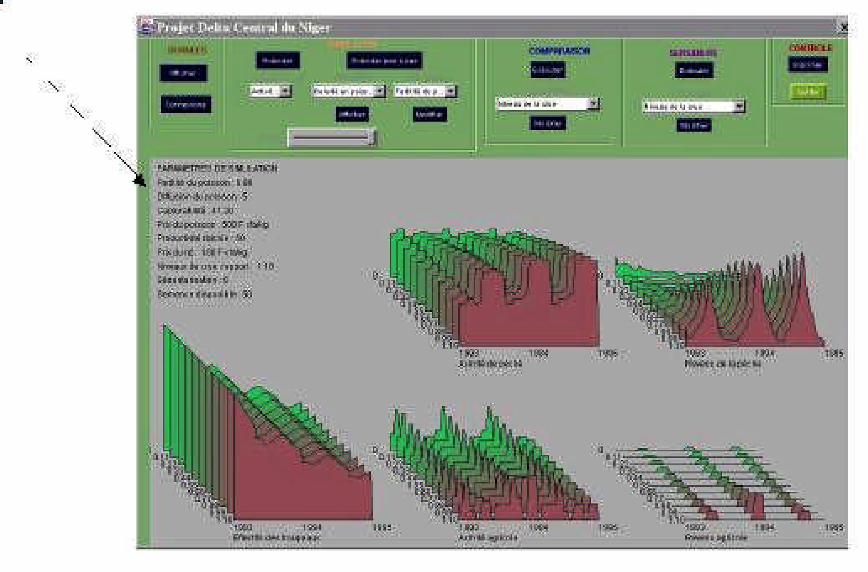


Analyse de sensibilité

- Choisir le paramètre de contrôle
- Fixer les bornes du paramètre de contrôle
- Effectuer

Analyse de sensibilité

- Effectuer



3. Extraction of pertinent information from images using Computer Vision techniques

This section describes Computer Vision algorithms and their application to Niger Delta data.

Extraction automatique d'informations pertinentes des images : application de techniques de Vision par Ordinateur

Rédacteur : Olivier MONGA (INRIA)

1- Problématique

Dans le contexte du projet SIMES, les images sont des vecteurs d'information importants. En effet, leur analyse permet de localiser les zones d'intérêt et surtout de suivre leur évolution au cours du temps. Ces images se caractérisent par leur diversité : aériennes, satellitales, radar…, ainsi que par la masse de données parfois considérable qu'elles représentent : par exemple les images aériennes du Delta Central du Niger (voir WP1) représentent 50 images de taille 11000 par 11000. La technologie des capteurs se développant de manière accélérée, ces deux caractéristiques : multi-modalité et taille des données vont encore s'amplifier. Ainsi, si l'opérateur humain reste le plus adapté pour l'extraction d'informations qualitatives et sémantiques, il est parfois assez démuni pour l'extraction d'informations quantitatives (mesure de zones caractéristiques,…) et pour le traitement systématique de ces données (mosaiquage, recalage…) souvent nécessaires pour une interprétation plus intelligente. Le modèle de gestion intégré du Delta Central (voir description) réalisé dans le cadre du projet SIMES pourrait par exemple être encore plus développé, surtout pour son processus de mise à jour, en disposant de manière récurrente de plus d'informations issues des images. Il est donc important d'investiguer les outils les plus récents de Vision par Ordinateur qui peuvent permettre d'extraire de manière automatique ou semi-automatiques des informations pertinentes des images.
De manière à réaliser ce transfert de l'expertise Vision par Ordinateur vers nos applications nous avons effectué les étapes suivantes :

- Expression des besoins des utilisateurs pour le traitement et l'analyse des images
- Identification des classes d'algorithmes de Vision par Ordinateur susceptibles d'être utiles
- Détermination de bibliothèques de logiciels libres, récentes et mises à jour correspondant à ces classes
- Sélection d'algorithmes pertinents par rapport aux besoins des utilisateurs
- Test de ces algorithmes sur des données du projet (couverture aérienne IGN du Delta central du Niger acquise par le projet et mise à disposition pour tous les partenaires par un site ftp - voir rapport d'activité 1998 -)

2- Besoins des utilisateurs

Les informations pertinentes pour les thématiciens (géographes, halieuthes, pédologues…) dans les images sont principalement :

- Les zones caractéristiques pour la modélisation de l'Ecosystème (voir modèle de gestion intégré du Delta Central) ; par exemple pour l'opération pilote Delta Central du Niger :

- les rizières : identification et mesure des superficies - *agriculture* -

- les surfaces inondées et les eaux libres (sans végétation par dessus) : localisation et mesure des superficies - *pêche* -

- les pâturages secs (petits arbustes) : identification et mesure de la superficie - *élevage et agriculture* -

- les ligneux (arbres, buissons, végétations)

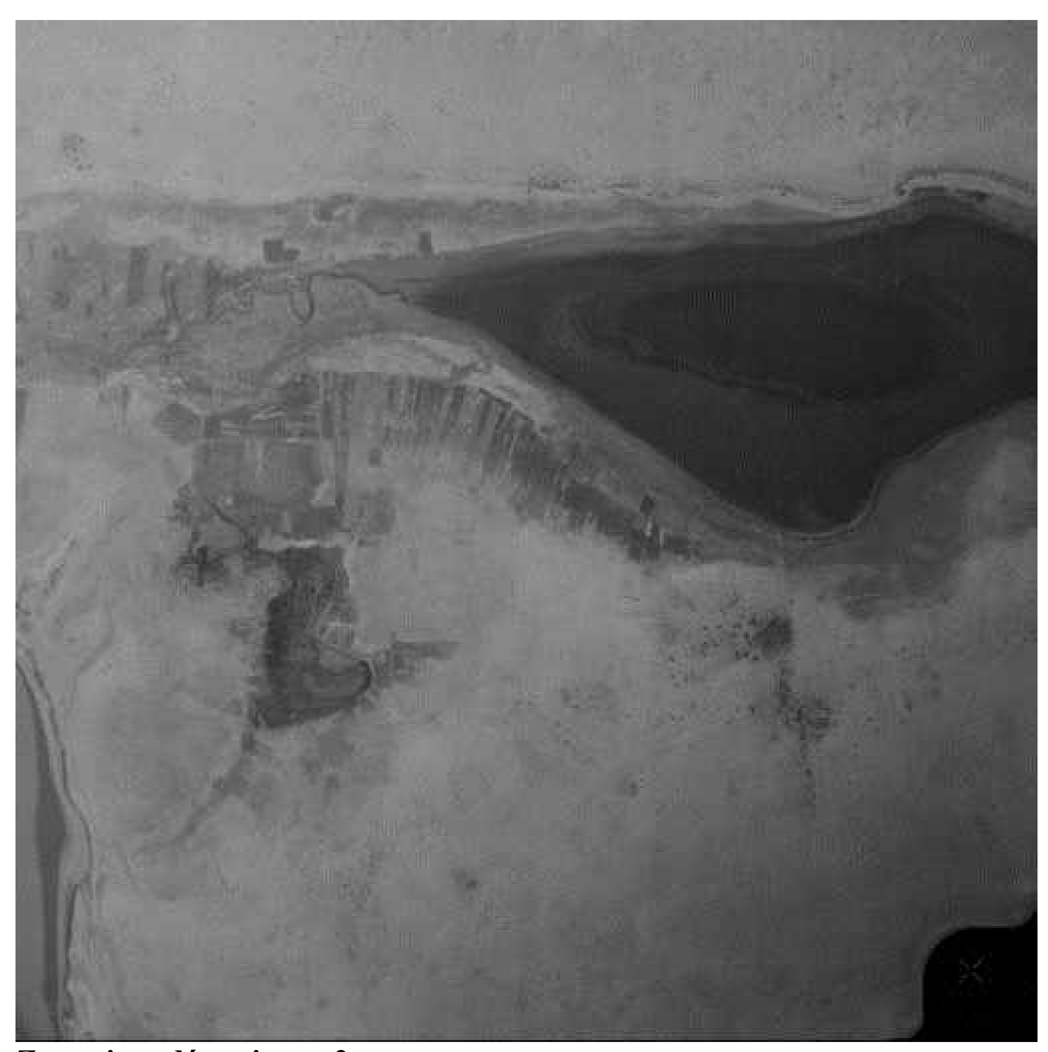
- La possibilité de mosaiquer (mettre dans un même repère) des images d'une même zone correspondant à un point de vue ou à des modalités différentes ; à l'heure actuelle cette opération est réalisée principalement manuellement par les thématiciens de l'IRD.

- La reconstruction de modèles numériques de terrains ; cette tache est bien sur intéressante si le relief est significatif ce qui n'est pas trop le cas pour notre opération pilote Delta Central du Niger.

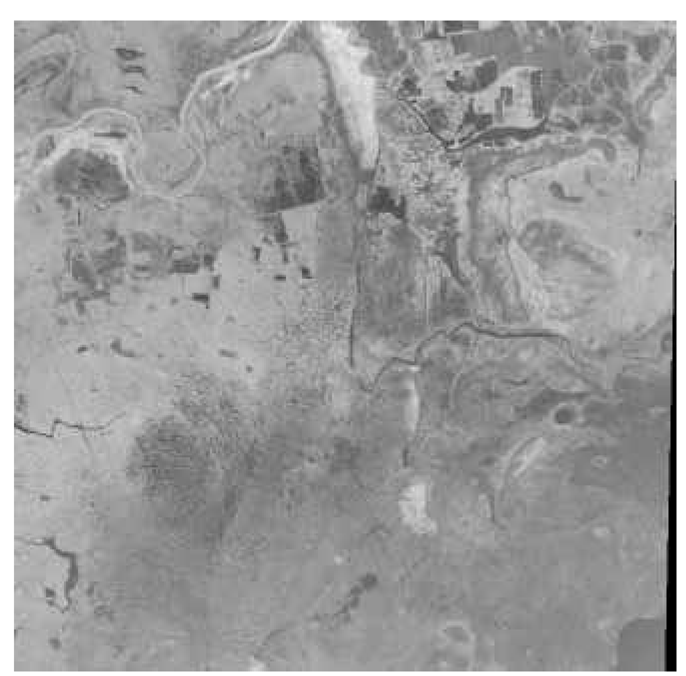
Ci dessous quelques images qui illustrent les points précédents. Ces images sont extraites de la couverture aérienne du Delta Central du Niger réalisée par l'IGN.



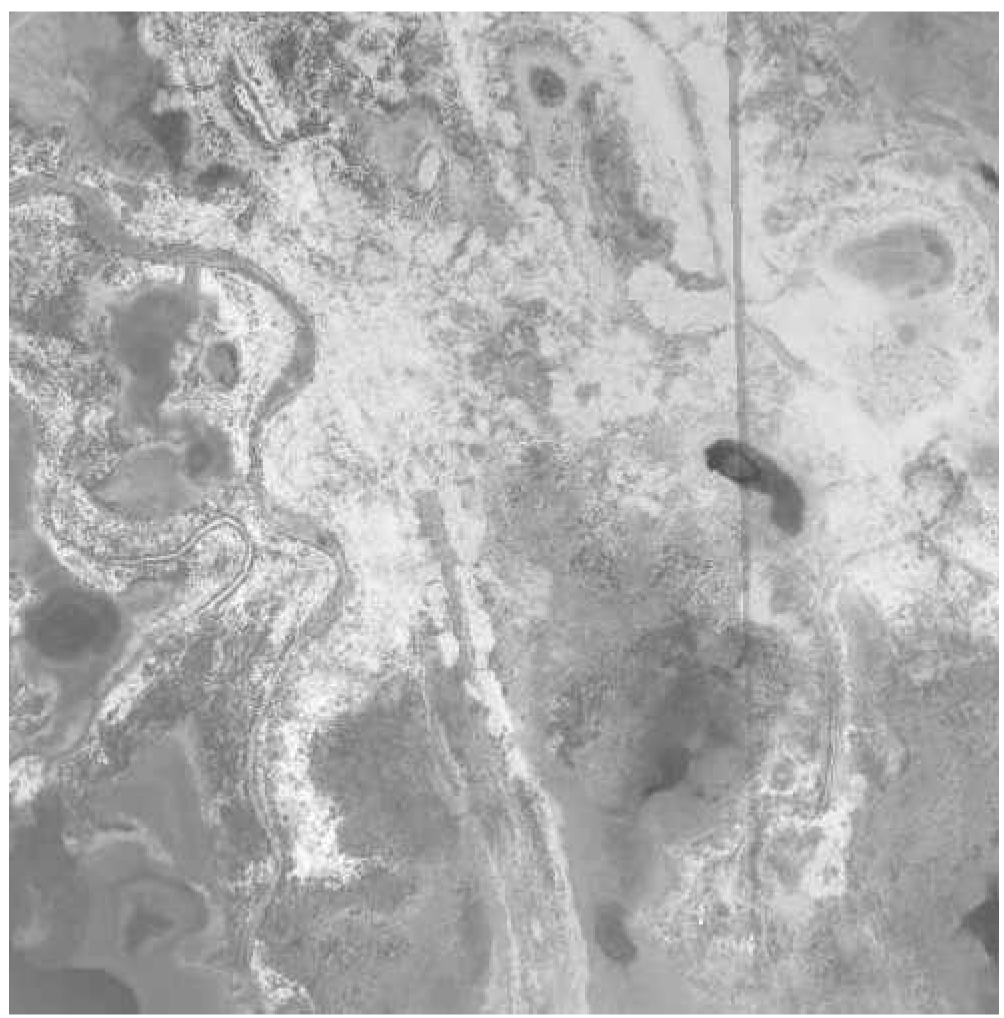
Rizières : image1



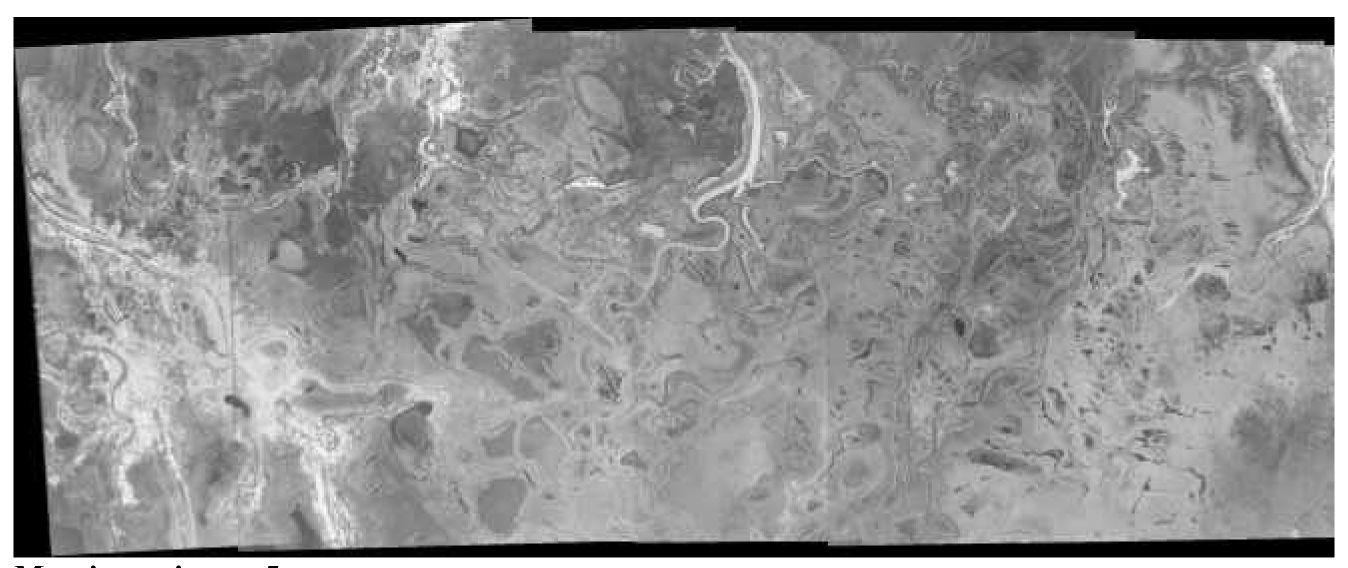
Zones inondées : image2



Pâturages secs : image 3



Ligneux : image 4



Mosaique : image 5

3- Classes d'algorithmes de Vision par Ordinateur

Les taches mentionnées dans le paragraphe ci dessous concernent deux parties de la Vision par ordinateur correspondant à deux grandes classes d'algorithmes :

- la Vision Précoce qui consiste à segmenter les images en zones caractéristiques du point de vue de la distribution des niveaux de gris (détection de contours, extraction de points caractéristiques, partition en régions, segmentation en texture….)
- La Vision Tridimensionnelle qui consiste à extraire des informations 3D à partir d'images 2D ; en particulier les algorithmes les plus performants de mosaiquage d'images utilisent des méthodes directement issus de la Vision Tridimensionnelle.

Il est donc intéressant pour SIMES de disposer des algorithmes les plus récents de Vision Précoce et de Vision Tridimensionnelle et de pouvoir les mettre à disposition.

4- Megawave2 et Targetjunior

4.1 Introduction

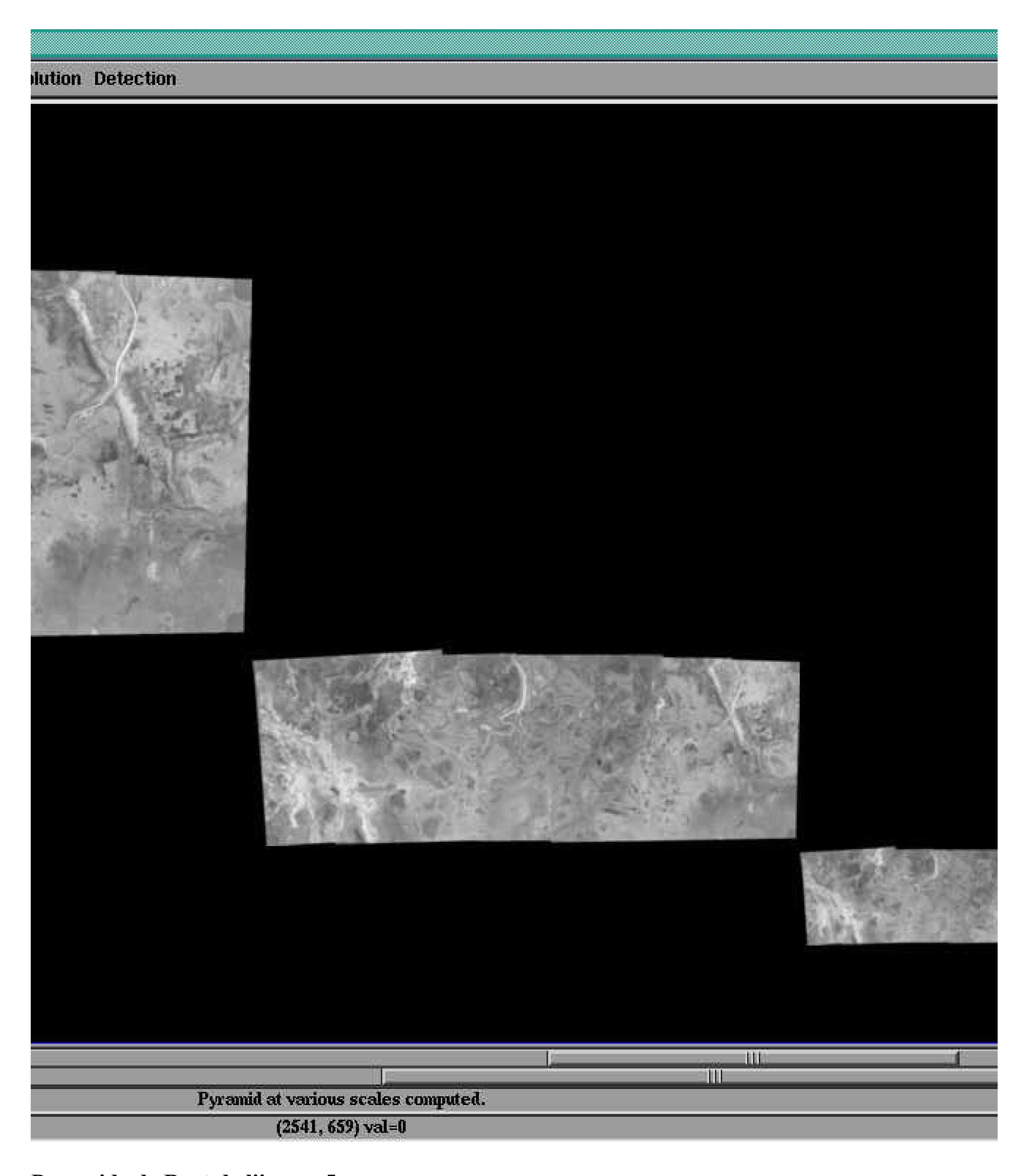
On a choisi d'utiliser pour le projet SIMES et donc d'installer chez nos partenaires africains (ESP-Dakar, UDS-Dshang, ESI-Bobo, CNTIG-Abidjan, IER-Bamako) les logiciels libres (sources compris) Targetjunior (ftp://ftp.esat.kuleuven.ac.be/pub/psi/visics/TargetJr/) et Megawave2 (http://www.cmla.ens-cachan.fr/Cmla/Megawave/index.fr.html). Les raisons principales de ce choix sont que ces deux plates-formes, développées par les partenaires du projet SIMES (Targetjunior) et par des laboratoires proches (Megawave2), incluent les outils les plus récents ainsi que les algorithmes de base de Vision Précoce et de Vision Tridimensionnelle. Elles constituent donc un état de l'art logiciel très à jour de l'existant en Vision par Ordinateur susceptible d'être utilisé dans le cadre de SIMES.

4.2 Targetjunior

Targetjunior résulte d'une initiative toujours active prise dans les années 1990 entre des laboratoires de recherche en Vision par Ordinateur de l'université d'Oxford, de l'INRIA, de l'Université de Louvain, et de General Electric (USA). Targetjunior est maintenu à l'Université d'Oxford et à l'Université de Louvain et constitue à l'heure actuel le logiciel libre de Vision le plus complet. Ce logiciel a été utilisé et développé dans le cadre de plusieurs projets européens : ESPRIT longterm research 23515, INPROOFS (aide aux criminologues avec le traitement d'images). Il constitue à la fois un outil de démonstration, un outil de développement et un outil d'enseignement (utilisé à Louvain comme outil d'enseignement). Il comprend 2000 fonctionnalités et représente 500 mégas compilé. Une personne (Peter Van Roose) travaille à temps plein pour sa maintenance à l'Université de Louvain. Les fonctionnalités 2D et 3D de Targetjunior couvrent tous les domaines du traitement d'images. Targetjunior est installable sur un portable Linux et sur toute plateforme et compilable sur toutes les architectures.

Indépendamment de l'aspect générique de Targetjunior intéressant pour l'exploitation long terme de SIMES et pour le transfert d'expertise Informatique-Thématiciens et Nord-Sud, nous avons choisi ce logiciel essentiellement pour ses fonctionnalités en Vision Précoce "première génération" (filtrage linéaire, morphologie mathématique…) et en Vision Tridimensionnelle (recalage, vision stéréoscopique…). Targetjunior inclut en particulier les algorithmes les plus récents de Vision Tridimensionnelle pour lesquels l'INRIA et l'Université d'Oxford sont parmi

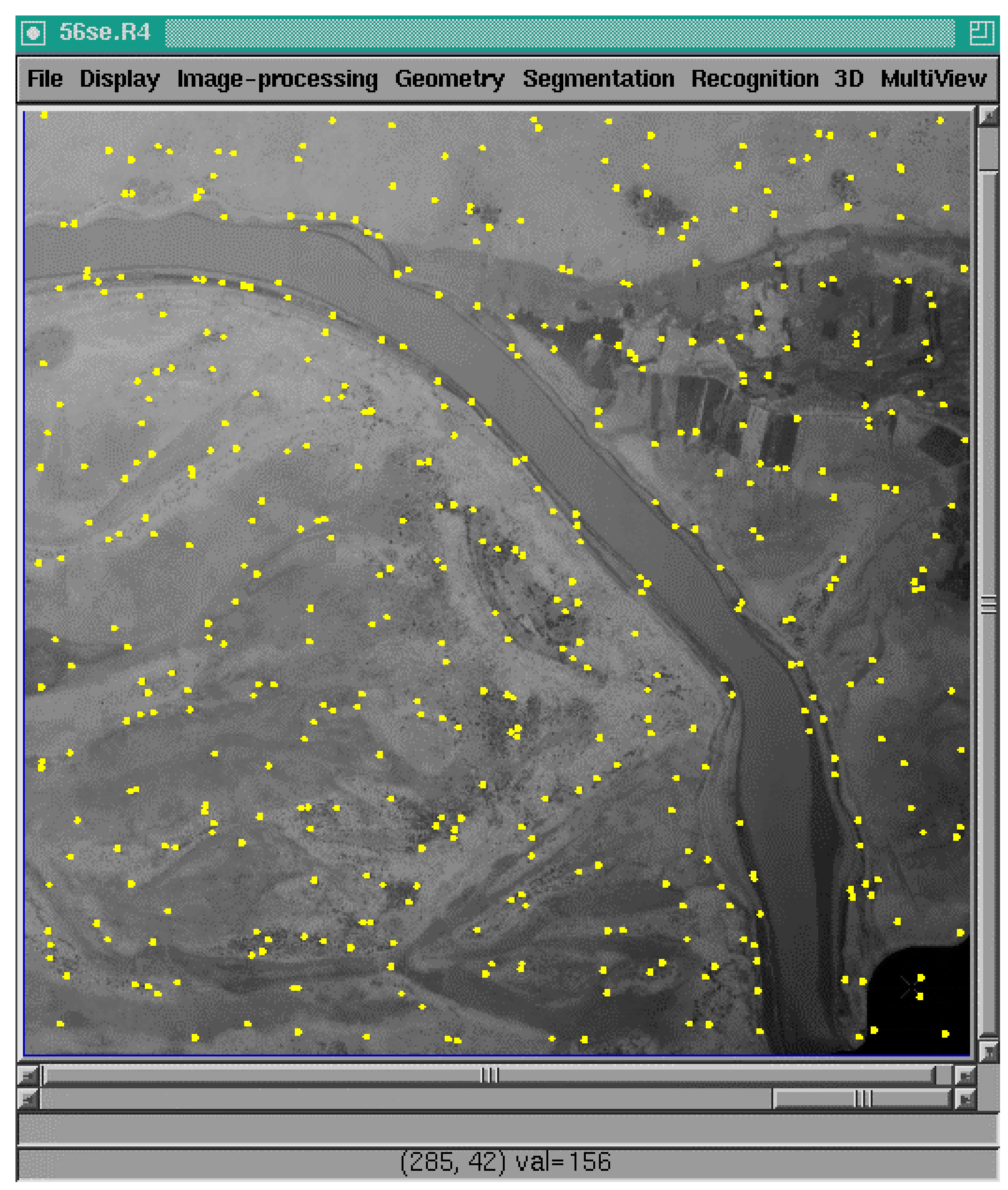
les tous premiers leaders mondiaux. Par ailleurs, il comprend aussi une bibliothèque assez complète d'algorithmes de segmentation d'images classiques et néanmoins actuels basés sur le filtrage auquels s'adjoignent les traitements de base de la morphologie mathématique.
Les images suivantes montrent des exemples de résultats obtenus sur nos données avec Targetjunior.



Pyramide de Burt de l'image 5



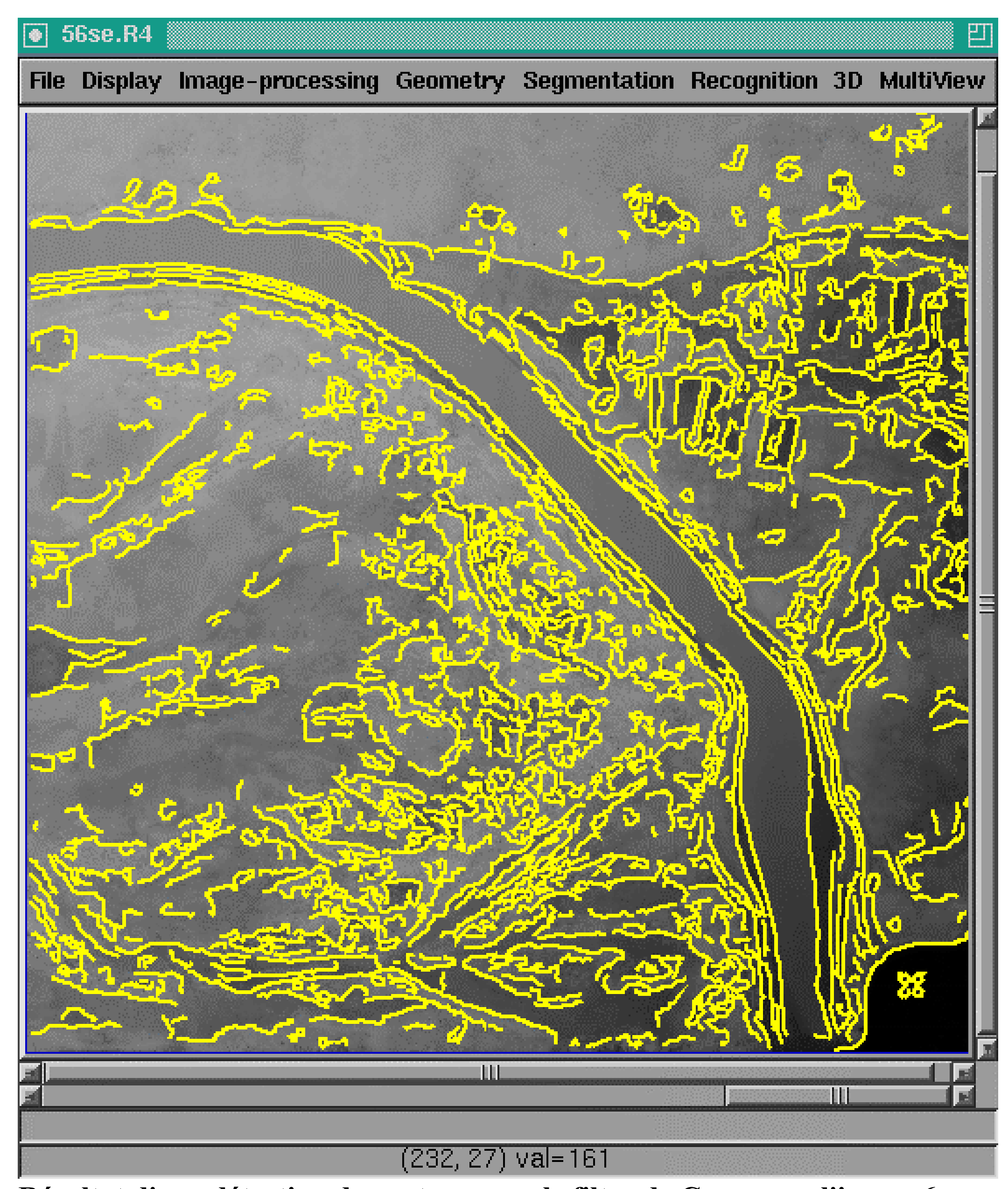
Bras du fleuve Niger : image 6



Résultat du détecteur de coins de Harris sur l'image 6 (ces points caractéristiques sont utilisés pour le mosaiquage automatique)



Résultat d'une ouverture-fermeture (morphologie mathématique) sur l'image 6



Résultat d'une détection de contours par le filtre de Canny sur l'image 6

4.2 Megawave2

Megawave2 est développé depuis 1993 par l'équipe de recherche dirigé par Jean Michel Morel (CEREMADE puis ENS-Cachan) et constitue une bibliothèque incluant principalement les algorithmes les plus récents de segmentation d'images basés sur les Equations aux Dérivées Partielles (EDP). Les EDPs ont commencé à être utilisées pour la segmentation d'images dans les années 1990 (Mumford, Shah, Morel, Faugeras, Deriche…) et ont permis l'introduction d'algorithmes très performants assez complémentaires des approches plus classiques. Notamment pour la segmentation en régions homogènes au sens des niveaux de gris ou des textures, les méthodes reposant sur les EDP sont très efficaces.

Ainsi, Megawave2 est complémentaire de Targetjunior pour les applications liées à SIMES. De plus Megawave2, est maintenu par l'ENS-Cachan qui collabore de longue date avec les équipes de l'INRIA.

5- Sélection d'algorithmes pertinents et résultats expérimentaux

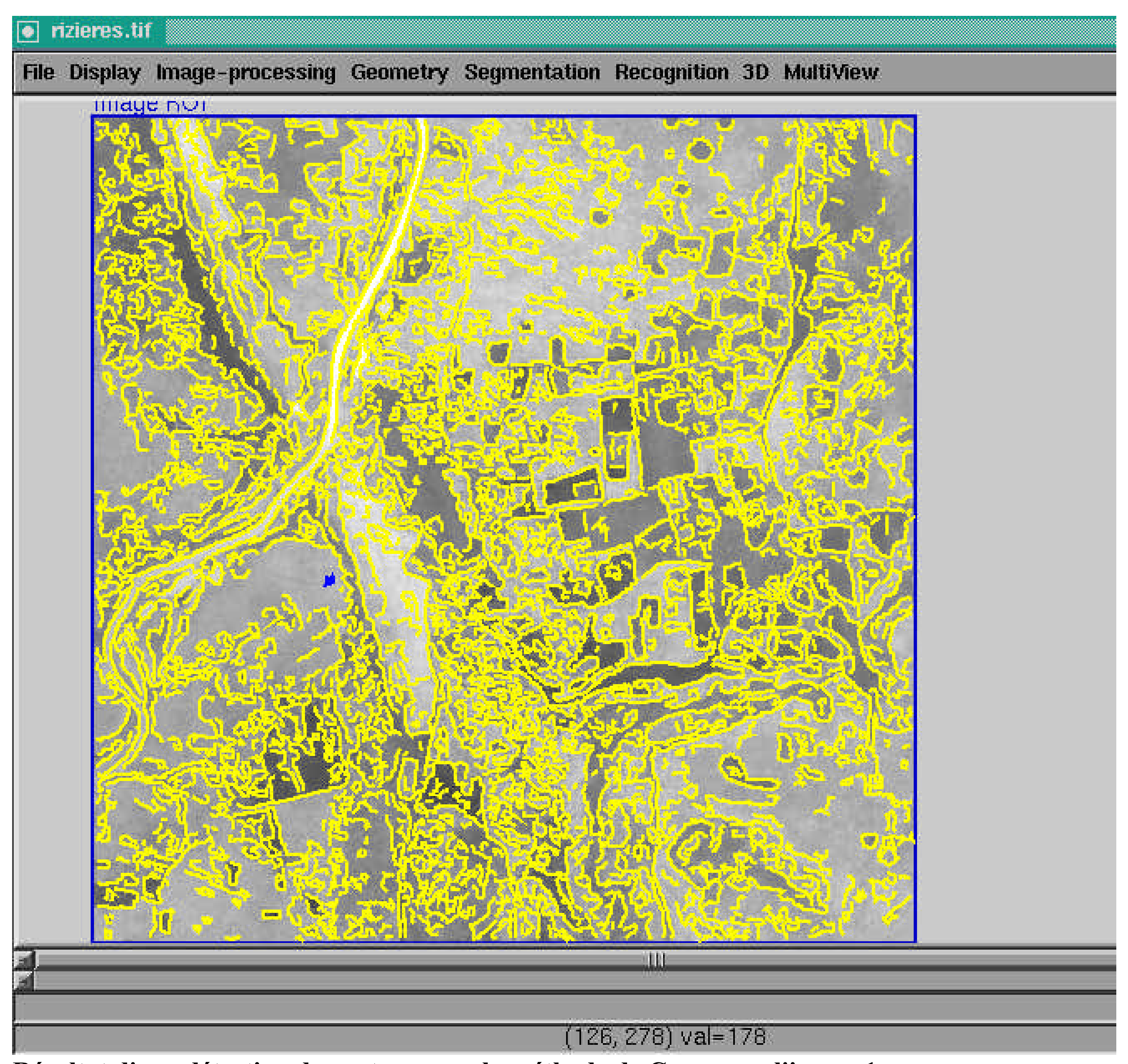
Nous avons sélectionné des algorithmes de Targetjunior et de Megawave2 pertinents pour les utilisateurs car permettant l'extraction automatique des informations décrites dans la section 2 ("Besoins des utilisateurs").

5.1 Extraction des zones de rizières

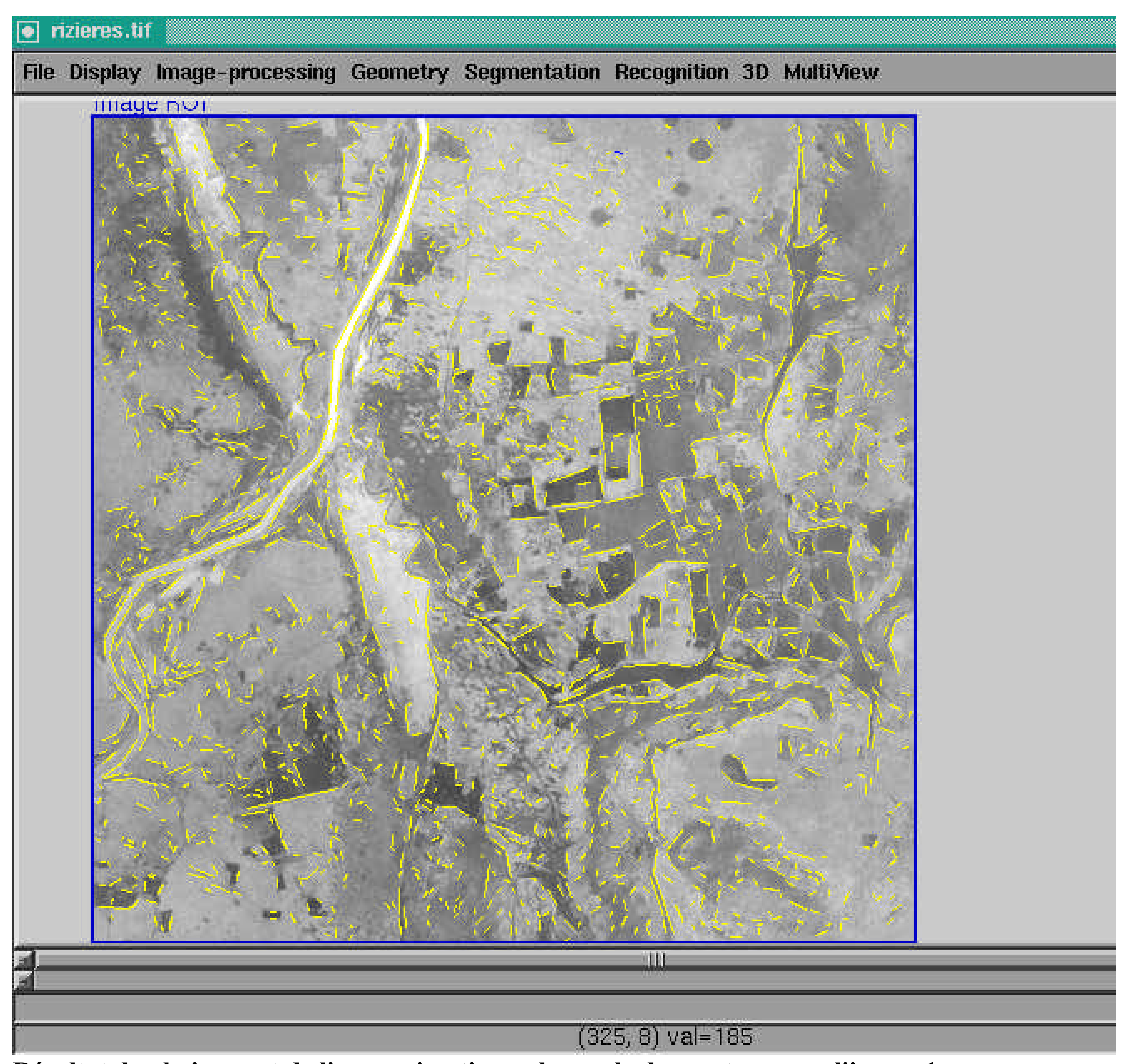
L'extraction automatique des zones de rizières est réalisée avec la suite d'algorithmes suivants:

- calcul du gradient de l'image par le filtre de Canny (voir état de l'art des algorithmes de traitement d'images section 1.4.3) - *Targetjunior* -
- extraction des maxima locaux de la norme du gradient dans la direction du gradient
- seuillage par hysteresis en fonction de la norme du gradient pour éliminer les point de norme de gradient trop faible - *Targetjunior* -
- chaînage des points de contour pour obtenir des ensembles connectés de points de contour - *Targetjunior* -
- approximation polygonale des chaines de contours - *Targetjunior* -

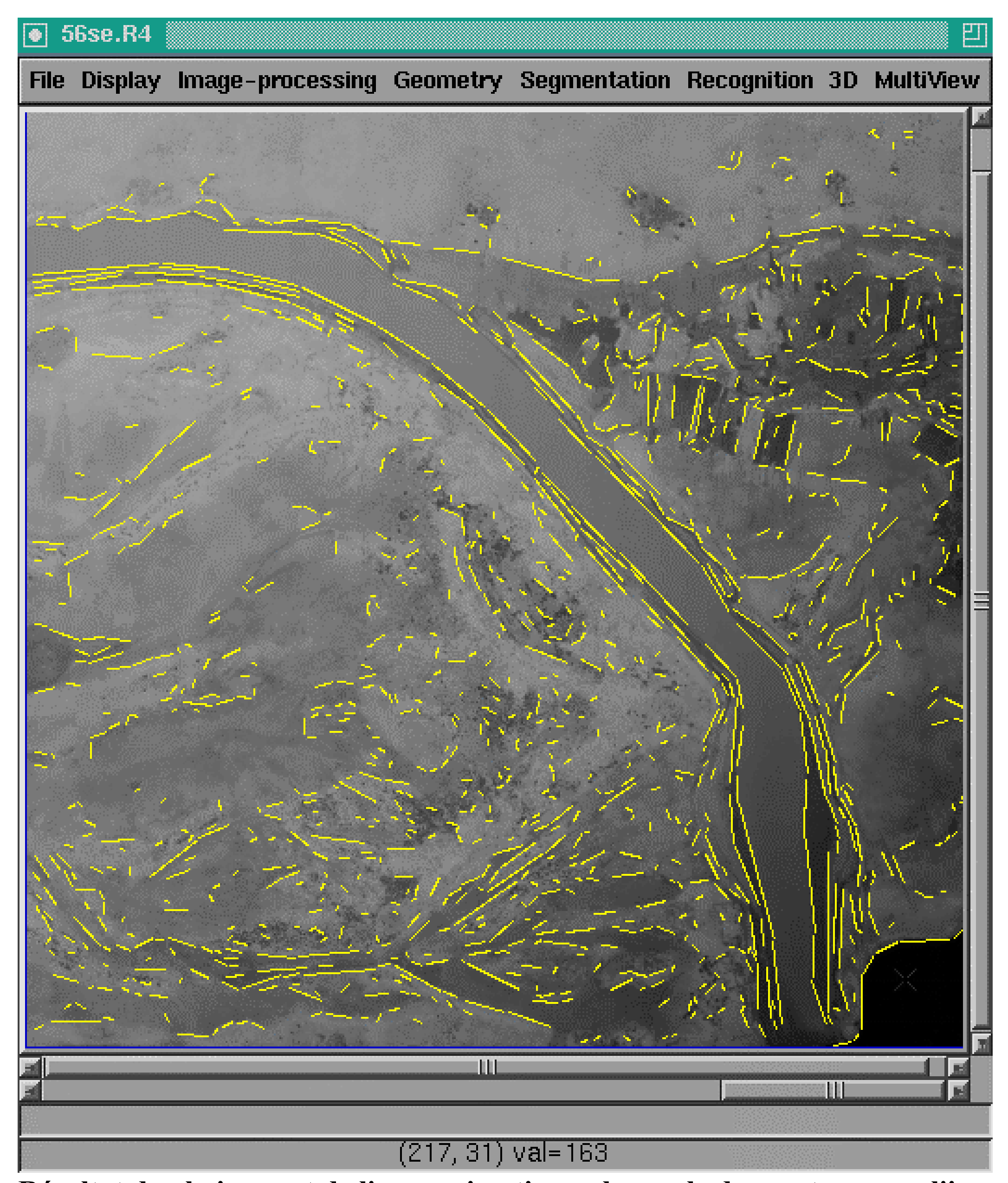
Les images suivantes montrent des résultats de traitements.



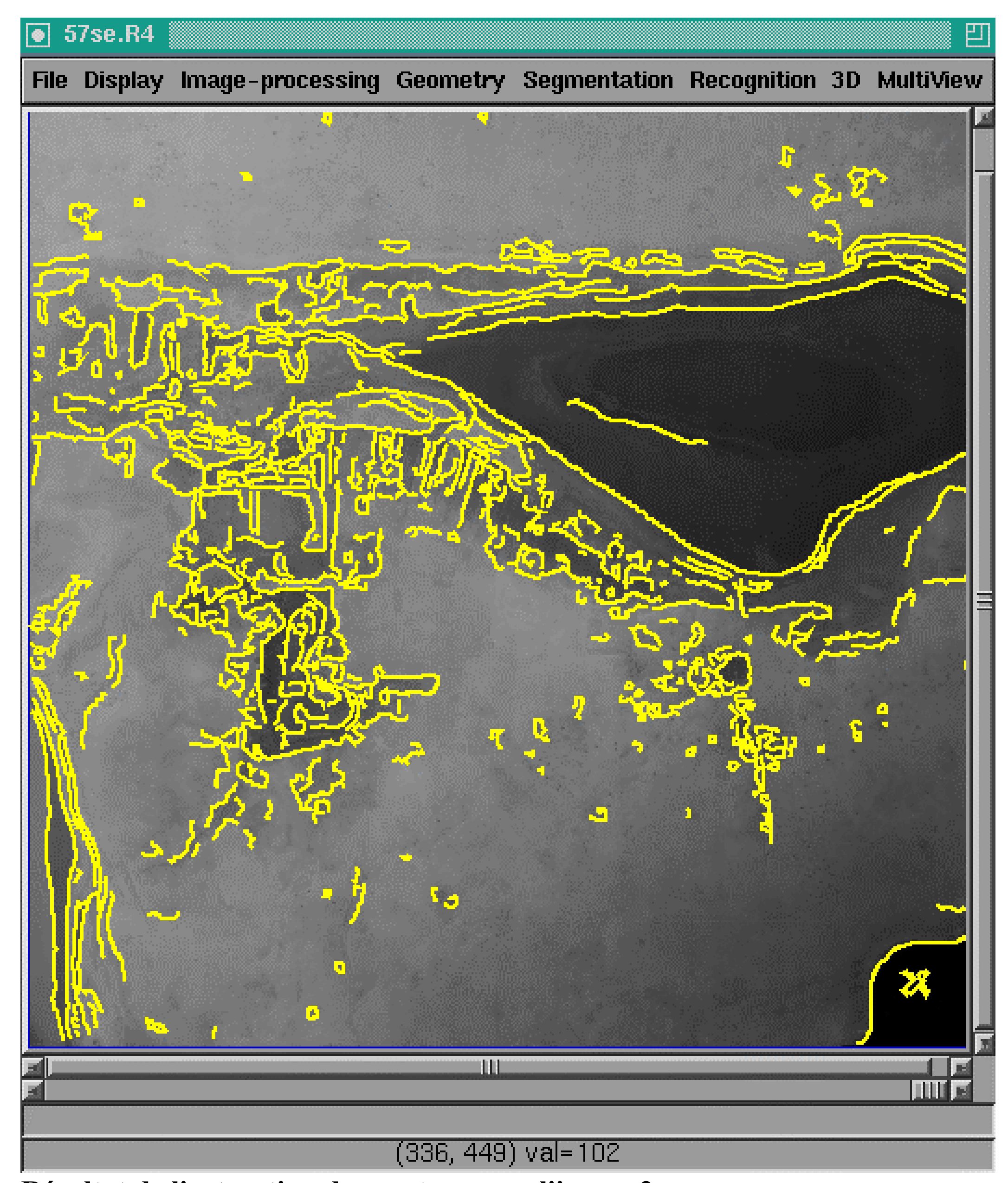
Résultat d'une détection de contours par la méthode de Canny sur l'image 1



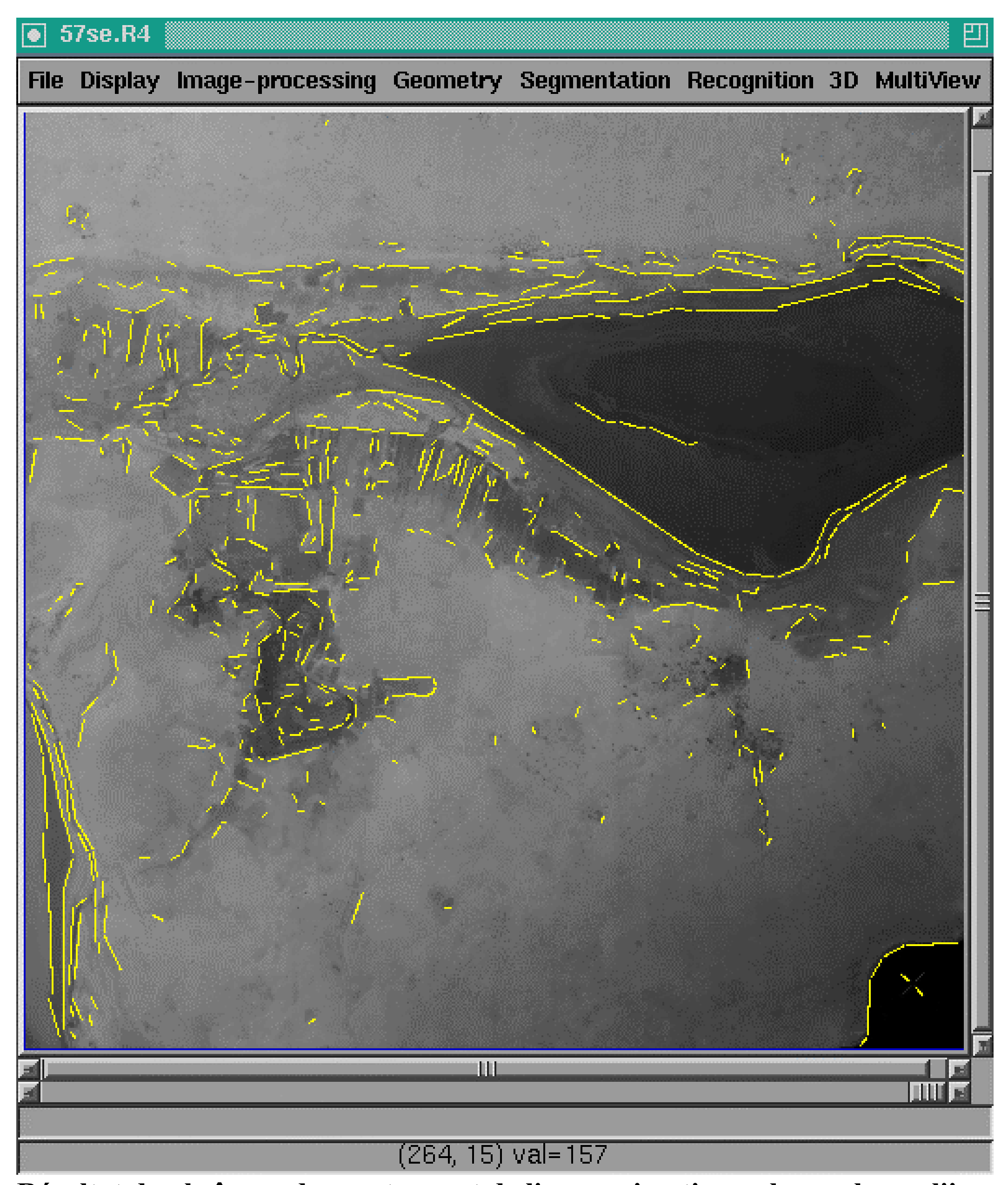
Résultat du chainage et de l'approximation polygonale des contours sur l'image 1



Résultat du chainage et de l'approximation polygonale des contours sur l'image 6



Résultat de l'extraction des contours sur l'image 2



Résultat du chaînage des contours et de l'approximation polygonale sur l'image 2

Le résultat final obtenu est un ensemble de segments de contours. La détermination des zones de l'image contenant des segments orthogonaux et parallèles permet de détecter les rizières. Ce traitement sera décrit dans le déliverable D13 ("Parameters extraction").

Une autre alternative pour effectuer l'extraction automatique des rizières est d'utiliser l'algorithme de croissance de régions par optimisation de fonctionnelle de David Mumford (Médaille Fields en géométrie algébrique) et Shah. Le principe de cette méthode est de déterminer le partitionnement de l'image minimisant un critère composite comprenant l'addition d'un terme définissant l'homogénéité des régions (somme des écarts quadratiques à la moyenne) et de la somme des longueurs des frontières (multiplié par un paramètre Lambda). Cet algorithme donne des résultats comparables aux techniques de region growing (voir état de l'art 1.3.2) avec une plus grande facilité d'utilisation (un seul paramètre) du à une formalisation mathématique claire de ce qui est calculé.

Le module de Megawave2 correspondant est "segct" . Ce module appartient à la partie "segmentation d'images en régions" de Megawave2 dont la description des modules suit.

○Name

mschannel Build a multi-scales multi-channels decomposition of an image

○Command Synopsis

mschannel [-N *N*] [-S *S*] [-W *W*] [-p *p*] *in fmovieout*

-N *N* : Number of images per channel - involved in the local scale value
-S *S* : Sigma - Standard deviation of the smoothing filter
-W *W* : Weight of the considered pixel in the smoothing filter
-p *p* : - 1 for ABS - 2 for Quadratic difference, default 2
in : input Fimage
fmovieout : output Fmovie

○Function Summary

void mschannel (N , S , W , p , in , fmovieout)
int *N , *W , *S ;
float *p ;
Fimage in ;
Fmovie fmovieout ;

○Description

This module build a multi-scales multi-channels representation of an image.
The aim of the algorithm is to create channels so as they can be used with *msegct* module in order to find the segmentation. We consider three kind of channels. Each channel correponds to a direction of a quadratic difference if p is set to 2 (default) or a simple difference in absolute value if *(p=1)*. More over we consider different scale for each channel.
Let e the local scale, and N the number of channels per direction that the user wants to reach. So for each local scale we have three channels. Let $I_{m,n}$ the value of the original image's pixel (input Fimage). From the original image we calculate the channels (Fimage) associated with the value of the local scale :
For $e = 1$ we have :

$$\text{Horizontal Channel : } H_{m,n}^{(e=1)} = (abs(I_{m+1,n} - I_{m,n})^p + abs(I_{m-1,n} - I_{m,n})^p)/2$$
$$\text{Vertical Channel : } V_{m,n}^{(e=1)} = (abs(I_{m,n+1} - I_{m,n})^p + abs(I_{m,n-1} - I_{m,n})^p)/2$$
$$\text{Diagonal Channel : } V_{m,n}^{(e=1)} = (abs(I_{m+1,n+1} - I_{m,n})^p + abs(I_{m-1,n-1} - I_{m,n})^p)/2$$

Those three channels correspond to the 3 images of the fmovie associeted with the first local scale $e = 1$. Then we compute the other channels belonging to the upper odd local scale. We need odd local scale because now we blur the input fimage. This is the multi-scale part of the algorithm. So we compute for $e > 1$ and odd :

$$\text{Horizontal Channel : } H_{m,n}^{(e>1)} = (abs(\hat{I}^e_{m+e,n} - \hat{I}^e_{m,n})^p + abs(\hat{I}^e_{m-e,n} - \hat{I}^e_{m,n})^p)/2$$
$$\text{Vertical Channel : } V_{m,n}^{(e>1)} = (abs(\hat{I}^e_{m,n+e} - \hat{I}^e_{m,n})^p + abs(\hat{I}^e_{m,n-e} - \hat{I}^e_{m,n})^p)/2$$
$$\text{Diagonal Channel : } V_{m,n}^{(e>1)} = (abs(\hat{I}^e_{m+e,n+e} - \hat{I}^e_{m,n})^p + abs(\hat{I}^e_{m-e,n-e} - \hat{I}^e_{m,n})^p)/2$$

$\hat{I}^e$ is the original fimage convolved with splitable blur filter of size $e * e$.
At least, all the channels are smoothed with a type of binomial filter. For that reason, the user needs to enter the standard deviation S and the weight W of the centered pixel . This filter is splitable and iterated $(2 + W) * S^2$ times.

○See Also

fsepconvol , fsmooth .
segtxt .

○Version 1.2

Last Modification date : Tue Dec 28 16:22:14 1999

○Author

Yann Guyonvarc'h

○Name

msegct Region-Growing method using the energy model of Mumford and Shah with piecewise constant approximation function, any number of channels

○Command Synopsis

msegct [-w *w_of_channels*] [-S *size_of_grid*] [-N *nb_of_regions*] [-L *lambda*] [-c *curves*] [-r *reconstruction*] *fmovie boundary*

-w *w_of_channels* : weights of channels, MW2-fsignal formated, (default weights 1/number_of_channels)
-S *size_of_grid* : size of initilization grid (int), default 1
-N *nb_of_regions* : number of desired regions (int)
-L *lambda* : value of final scale parameter (float) of last 2-normal segmentation
-c *curves* : output boundary set in curves format
-r *reconstruction* : output piecewise constant reconstruction of each channel
screen output : final number of regions
screen output : final lambda value
fmovie : original multichannel data in fmovie format
boundary : b/w image of boundary set

○Function Summary

Cimage msegct (weight , sgrid , nb_of_regions , lambda , curves , u , f_nb_of_regions , f_lambda , orig_data)
Fsignal weight ;
int *sgrid , *nb_of_regions , *f_nb_of_regions ;
float *lambda , *f_lambda ;
Curves curves ;
Fmovie orig_data , u ;

○Description

The command **msegct** computes a segmentation of multichannel data. Each channel is an `fimage` which is part of the input *fmovie*. Thus we consider a picture where to each pixel are associated `nb_ch`(`float`-type) numbers. The number of channels, `nb_ch`, is equal to the number of images of is which made *fmovie* (see how an `fmovie` is loaded in MW2).
The algorithm computes 2-normal segmentation of this picture as will be described below.
The initial picture is broken in a grid of elementary cells, composed of $D \times D$-pixel squares (D=*size_of_grid*). We use a classical region-growing algorithm to achieve the partition of the image in homogeneous regions.

The criterion which measures the resemblance of regions is given by the Mumford and Shah model. Call g the picture defined on an open rectangle $R \subset \mathbf{R}$, u a piecewise constant function, which approximates g, and let B be the boundaries between the regions, i.e. the set of jump points of u (remember that g and u are elements of $\mathbf{R}^{\texttt{nb_ch}}$). With Mumford and Shah we introduce the following functional which has to be minimized

$$\mathrm{E}(B) = \int_R \|u - g\|^2 + \lambda \ell(B),$$

where $\|.\|$ is the (weighted) norm we put on g-space:

$$\|u - g\|^2 = \sum_{i=1}^{\texttt{nb_ch}} w_i \cdot (u_i - g_i)^2,$$

with $w = (w_1, \ldots, w_{\texttt{nb_ch}})$ the weights the user puts on the channels. Moreover ℓ is the 1-dimensional Hausdorff measure and λ a real scaling coefficient. The parameter λ gives a weight to the length of the boundaries: a small value allows a lot of boundaries whereas a big value tends to reduce the boundary length $\ell(B)$.

The property of being 2-normal for a segmentation of R in regions $\bigcup_i O_i$ has been used by Pavlidis in the seventies. We say that B is 2-normal if, given any two regions O_i, O_j having common boundary $\partial(O_i, O_j)$, the following inequality for the energy holds

$$\mathrm{E}(B) \leq \mathrm{E}(B \setminus \partial(O_i, O_j)).$$

Which yields

$$\lambda \ell(\partial(O_i, O_j)) \leq \frac{|O_i| \cdot |O_j|}{|O_i| + |O_j|} \|u^i - u^j\|^2,$$

where $|.|$ denotes the surface measure and u^i, u^j are the mean values of g on O_i, O_j, for example $u^i = \frac{1}{|O_i|} \int_{O_i} g \ = (u_1^i, \ldots, u_{\texttt{nb_ch}}^i) \in \mathbf{R}^{\texttt{nb_ch}}$.

The algorithm proceeds as follows. The initial picture is partitioned in squares of side *size_of_grid* pixels. Construct a list of the corresponding symbolic regions containing all the information needed: surface, sum of gray-levels in the square (for each channel), length of boundaries. Construct a table in which to each boundary is associated the value of λ for which this boundary will disappear (i.e. E decreases by merging the two adjacent regions). Using and updating this "merit" table the algorithm proceeds. The information on the new region, constructed when a merging decision occurred, is taken from the two old one's, so there is no need to return on pixel level.

There are two (incompatible) options : either one wants the segmentation to stop at *nb_of_regions* regions or the final segmentation is at scale *lambda*.

If the first option is chosen the program stops when the desired number of regions (*nb_of_regions*) is reached. To be more precise it stops at the value of λ for which this particular number of regions is reached, there might be less regions remaining if *nb_of_regions* doesn't correspond to a 2-normal segmentation for any λ.

In case the final λ is fixed the program stops at *lambda* or for the value of λ just lower than *lambda*. For example if *lambda*=13.45 and the current λ_c=13.09, then if the next boundary in the segmentation will vanish for λ_n=14.67 the program stops. The obtained segmentation is also valid for each λ strictly lower than the next value λ_n.

If the program is used as module in another program the final values of λ and the number of regions are passed to the corresponding variables. On command line execution this information is anyway displayed.
The program estimates the memory used by the process, this mainly depends on the precision of the initial grid, for example $D = 1$ is the best possible precision, but it needs a lot of memory for the data structure (1 pixel = 1 region). So the program turns out to be most efficient (i.e. fast) on machines with a big amount of central memory (RAM). Indeed as there is no previous information on the picture available and as regions can grow in any direction, the whole structure should be available in central memory.
The output shows the initial quadratic error, the total boundary length and the number of regions the algorithm starts with, after reaching *nb_of_regions* the quadratic error, boundary length, number of region and the current value of λ.
The output is file *boundary* which represents the boundary set on a white background and is stored under `cimage`-format.
An optional output is file *curves* which contains the boundary set under `curve`-format (use **-c**). Using **curves_cimage** the result can be visulized on the original *fimage* for example.
To obtain the gray-level reconstruction of the piecewise constant approximation for each channel use `-r`. This will write an `Fmovie` in file *reconstruction*.

○See Also

segtxt .

○Version 1.31

Last Modification date : Tue Dec 21 17:09:11 1999

○Author

Georges Koepfler

○Name

one_levelset Get boundaries of level set, using a simplified merging criterion in the 'well-known' segmentation algorithm

○Command Synopsis

one_levelset [-l *level*] [-b *boundary*] [-p *polygons*] [-G *f_levelset*] [-B *b_levelset*] *fimage*

-l *level* : pixels ¡='level' (float) belong to the level set, default 127
-b *boundary* : output boundary of levelset, file cimage formated
-p *polygons* : output boundary of levelset, file fpolygons formated
-G *f_levelset* : output levelset with gray-'level', file fimage formated
-B *b_levelset* : output levelset b/w, file cimage formated
fimage : original image

○Function Summary

void one_levelset (level , cb , pb , fu , cu , image_org)
float *level ;
Cimage cb , cu ;
Fpolygons pb ;
Fimage fu ;
Fimage image_org ;

○Description

This function generates a level set out of *fimage*. More precisely the program segments the original image into two classes of regions. First the regions which have gray level lower or equal than *level*, the other regions are those with gray level above *level*.
In this application we consider the level set L to be given by

$$L = \{(x, y) \ / \ g(x, y) \leq \mathit{level}\}$$

where g is the original picture.
Different possibilities to view the result are given. Either one wants the boundary of the level set, the **-b** option draws the boundary set in a `cimage` *boundary* and **-p** writes the coordinates of the contours into `fpolygons` *polygons*. Or one wants an image of the level set, then with **-B** we obtain a black and white picture (the pixels of L being black) and with **-G** the pixels of L will be drawn in color $\sup\{g(x, y) \ / \ (x, y) \in L\}$ and the other pixels will be in color $\sup\{g(x, y) \ / \ (x, y) \notin L\}$. For example you want the *f_levelset* for *level*=127 of an image which has only gray values 0,20,50,100,200 and 220. Then the set L will be drawn with gray 100 and the other pixels with gray 220.

Notice that L is made of disjoined (in the 4-neighborhood sense) connected sets which thus have <u>closed</u> Jordan curves as boundaries.
Let us give some more details about the **-p** option. The file *polygons* will be in the `MW2`-format `fpolygons`. The coordinates of the boundary are `floats` as they always have a decimal part of 0.5 . In figure 3 we represent a image where the pixels are represented by squares (white or gray). The level set (gray squares) is bounded by its border, the black dots (•) which are drawn are the points you will find in the `fpolygons` structure, the coordinates can be read on the axes drawn above and besides the "pixels".

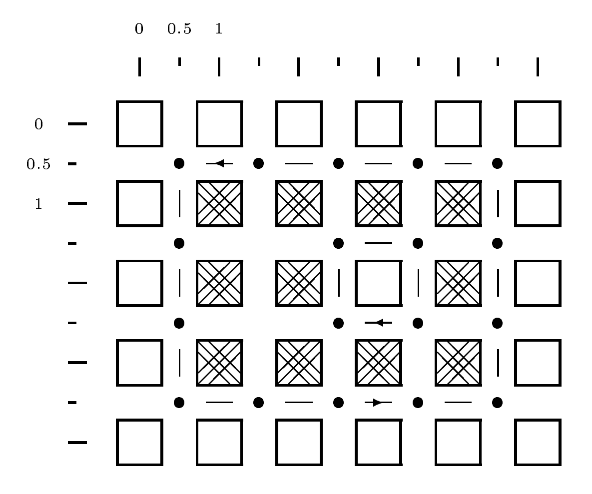


Figure 3: 6 × 5 pixels image.

Each `fpolygon` has two channels of information, the first is the *level*, the second is a signed label. This label (`int`) shows which contours belong to the same set, if the label is positive the current contour is the 'outer' boundary of the set (thus it is unique), if the label is negative the current contour is 'inside' the set (it is a hole). For example the image of figure 3 will yield one `fpolygon` labeled 1 with 14 points and one `fpolygon`, labeled -1, made out of 4 points only.
The contour is oriented (the pixels are ordered) such that the set is always on the <u>left</u> if you follow the list of points.
For example lets say that the result in *polygons* is made out of 5 `fpolygon` elements with labels 1,-1,2,-2,-2. This means that there are 2 connected sets which compose L, one having one hole the other one two. There is no information available whether set 1 is in a hole of set 2, or set 2 in the hole of set 1, or if both sets are completely apart.
Notice that if a set has just negative labeled contours then it is the background (as the image boundary is not coded).
Currently there are a "few" restrictions for the use of the **-p** option. The result of the segmentation should just contain regions which have only boundaries made out of one (1!) connected component. Also should a set of L either be the background or not touch at all the boundary of the image. If one of these cases occurs the program will exit the construction of the *polygons* file.

○Version 1.01

Last Modification date : Tue Dec 21 17:09:11 1999

○Author

Georges Koepfler

○Name

segct Region-Growing method using the energy model of Mumford and Shah with piecewise constant approximation function

○Command Synopsis

segct [-S *size_of_grid*] [-N *nb_of_regions*] [-L *lambda*] [-c *curves*] [-r *reconstruction*] *fimage boundary*

-S *size_of_grid* : size of initilization grid (int), default 1
-N *nb_of_regions* : number of desired regions (int)
-L *lambda* : value of final scale parameter (float) of last 2-normal segmentation
-c *curves* : output boundary set in curves format
-r *reconstruction* : output piecewise constant reconstruction
screen output : final number of regions
screen output : final lambda value
fimage : original image
boundary : b/w image of boundary set

○Function Summary

Cimage segct (sgrid , nb_of_regions , lambda , curves , u , f_nb_of_regions , f_lambda , image_org)
int *sgrid , *nb_of_regions , *f_nb_of_regions ;
float *lambda , *f_lambda ;
Fimage image_org , u ;
Curves curves ;

○Description

The command **segct** generates, starting from *fimage*, a 2-normal segmentation of this picture as will be described below.
The initial picture is broken in a grid of elementary cells, composed of $D \times D$-pixel squares (D=*size_of_grid*). We use a classical region-growing algorithm to achieve the partition of the image in homogeneous regions.

The criterion which measures the resemblance of regions is given by the Mumford and Shah model. Call g the picture defined on an open rectangle R, u a piecewise constant function, which aim is to approximate g, and let B be the boundaries between the regions, i.e. the set of jump points of u. With Mumford and Shah we introduce the following functional which has to be minimized

$$\mathrm{E}(B) = \int_R (u-g)^2 + \lambda \ell(B),$$

where ℓ is the 1-dimensional Hausdorff measure and λ a real scaling coefficient. The parameter λ gives a weight to the length of the boundaries: a small value allows a lot of boundaries whereas a big value tends to reduce the boundary length $\ell(B)$.

The property of being 2-normal for a segmentation of R in regions $\bigcup_i O_i$ has been used by Pavlidis in the seventies. We say that B is 2-normal if, given any two regions O_i, O_j having common boundary $\partial(O_i, O_j)$, the following inequality for the energy holds

$$\mathrm{E}(B) \leq \mathrm{E}(B \setminus \partial(O_i, O_j)).$$

Which yields

$$\lambda \ell(\partial(O_i, O_j)) \leq \frac{|O_i| \cdot |O_j|}{|O_i| + |O_j|} \|u_i - u_j\|^2,$$

where $|.|$ denotes the surface measure and u_i, u_j are the mean values of g on O_i, O_j, for example $u_i = \frac{1}{|O_i|} \int_{O_i} g$.

The algorithm proceeds as follows. The initial picture is partitioned in squares of side *size_of_grid* pixels. Construct a list of the corresponding symbolic regions containing all the information needed (surface, sum of gray-levels in the square, length of boundaries). Construct a table in which to each boundary is associated the value of λ for which this boundary will disappear (i.e. E decreases by merging the two adjacent regions). Using and updating this "merit" table the algorithm proceeds. The information on the new region, constructed when a merging decision occurred, is taken from the two old one's, so there is no need to return on pixel level.

There are two (incompatible) options : either one wants the segmentation to stop at *nb_of_regions* regions or the final segmentation is at scale *lambda*.

If the first option is chosen the program stops when the desired number of regions (*nb_of_regions*) is reached. To be more precise it stops at the value of λ for which this particular number of regions is reached, there might be less regions remaining if *nb_of_regions* doesn't correspond to a 2-normal segmentation for any λ.

In case the final λ is fixed the program stops at *lambda* or for the value of λ just lower than *lambda*. For example if *lambda*=13.45 and the current λ_c=13.09, then if the next boundary in the segmentation will vanish for λ_n=14.67 the program stops. The obtained segmentation is also valid for each λ strictly lower than the next value λ_n.

If the program is used as module in another program the final values of λ and the number of regions are passed to the corresponding variables. On command line execution this information is anyway displayed.

The program estimates the memory used by the process, this mainly depends on the precision of the initial grid, for example $D = 1$ is the best possible precision, but it needs a lot of memory for the data structure (1 pixel = 1 region). So the program turns out to be most efficient (i.e. fast) on machines with a big amount of central memory. Indeed as there is no previous information on the picture available and as regions can grow in any direction, the whole structure should be available in central memory.
The output shows the initial quadratic error, the total boundary length and the number of regions the algorithm starts with, after reaching *nb_of_regions* the quadratic error, boundary length, number of region and the current value of λ.
The output is file *boundary* which represents the boundary set on a white background and is stored under `cimage`-format.
An optional output is file *curves* which contains the boundary set under `curve`-format (use **-c**). Using **curves_cimage** the result can be visualized on the original *fimage* for example.
To obtain the gray-level reconstruction of the piecewise constant approximation use **-r**. This will write an `fimage` in file *reconstruction*.

○Version 1.21

Last Modification date : Tue Dec 21 17:09:11 1999

○Author

Georges Koepfler

○Name

segtxt Texture Segmentation using multi-scales multi-channels representation

○Command Synopsis

segtxt [-N *N*] [-S *S*] [-W *W*] [-p *p*] [-n *nr*] *in movieout out*

-N *N* : Number of images per channel - involved in the local scale value
-S *S* : Sigma - Standard deviation of the smoothing filter
-W *W* : Weight of the pixel in the smoothing filter
-p *p* : - 1 for ABS - 2 for Quadratic difference
-n *nr* : Number of desired regions
in : Input Fimage
movieout : output Fmovie
out : Output segmented Cimage

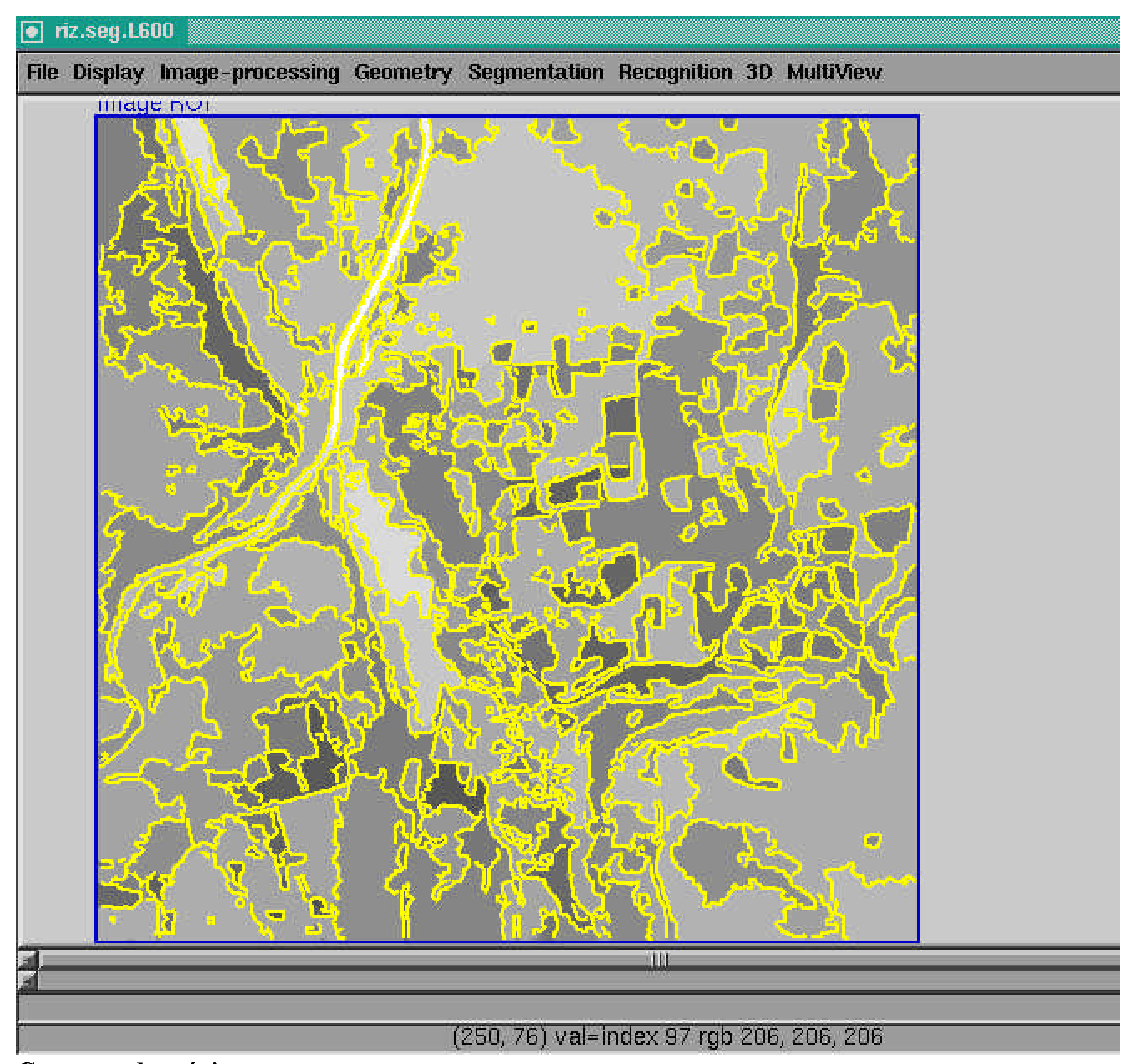
○Function Summary

```
void segtxt (N , S , W , p , nr , in , fmovieout , out )
int *N , *W , *nr , *S ;
float *p ;
Fimage in ;
Fmovie fmovieout ;
Cimage out ;
```

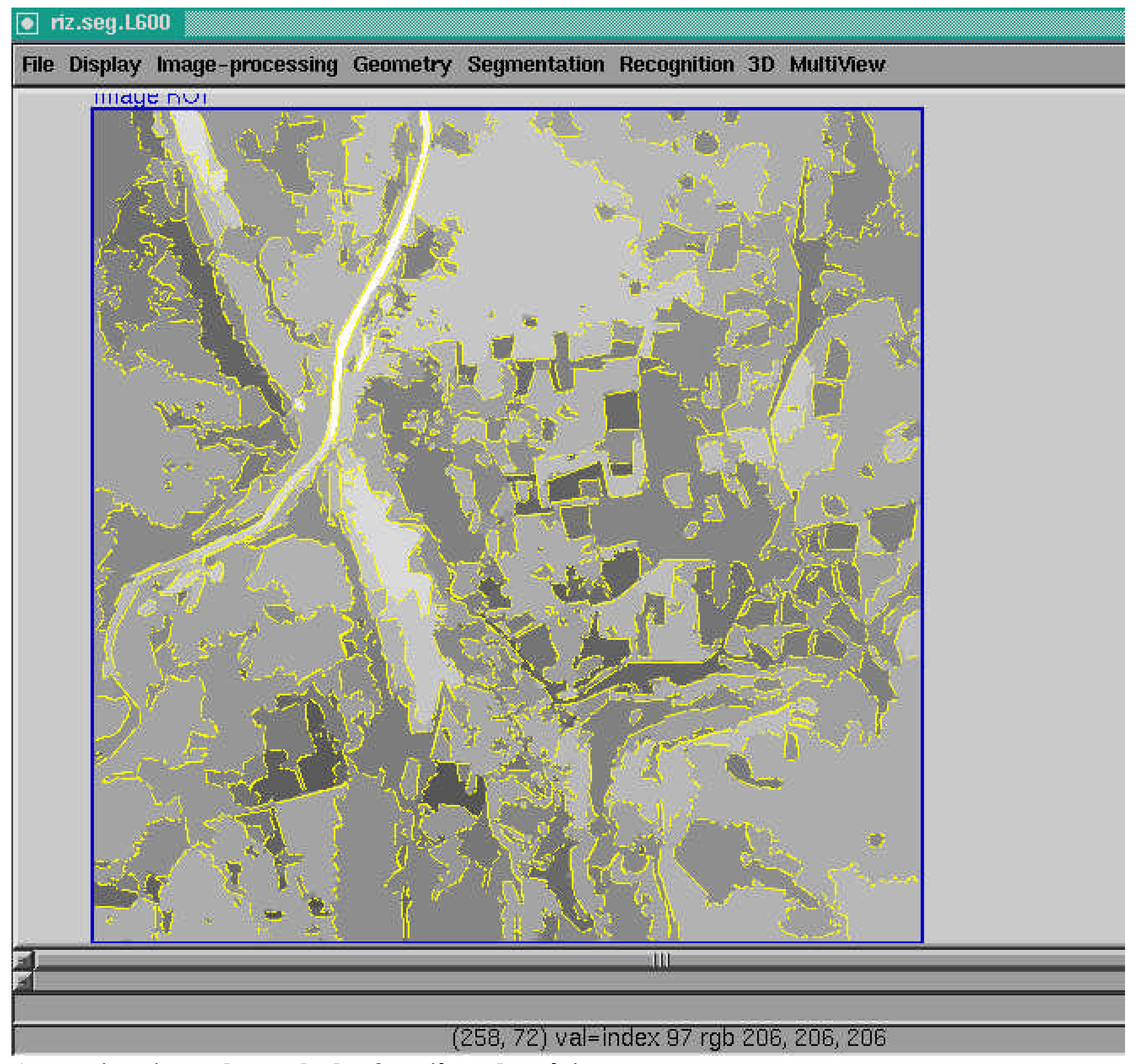
L'application de cet algorithme sur l'image 1 donne les résultats suivants :



Image des régions obtenues par l'utilisation de l'algorithme de Mumford-Shah sur l'image 1 (chaque région est coloriée avec sa moyenne de niveau de gris).



Contours des régions

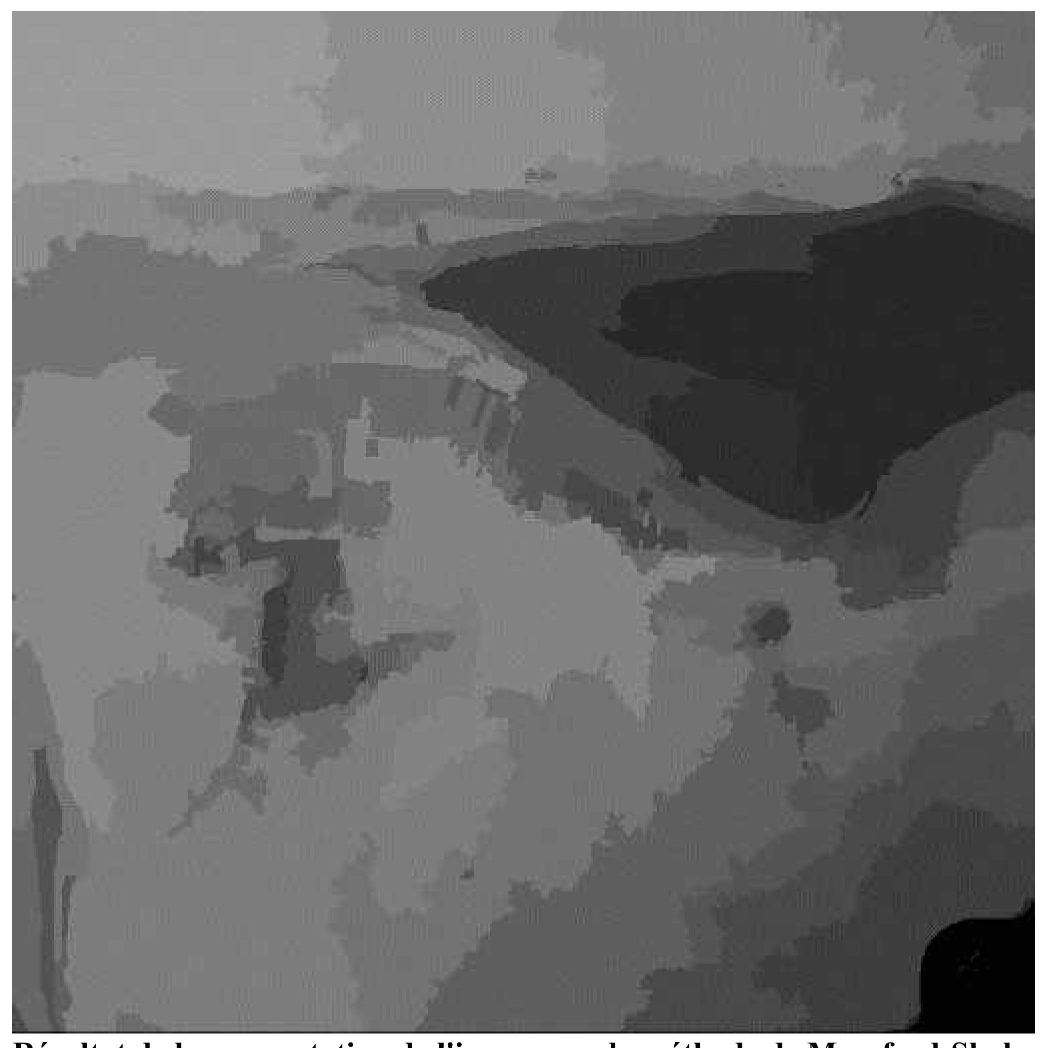


Approximation polygonale des frontières des régions

Ainsi, une détection des régions dont les contours sont parallélépipédiques permet aussi la détection des zones de rizières. Ce traitement venant après la segmentation en régions sera aussi décrit dans le déliverable D13.

5.2 Extraction de surfaces inondés et d'eaux libres

Les surfaces inondés et les eaux libres correspondent à des zones homogènes de l'image. On va donc là aussi utiliser la segmentation par optimisation de fonctionnelle décrite précédemment. Ensuite, on sélectionnera les zones homogènes non étiquetés rizières d'aire suffisamment importante (formes des frontières non rectilignes, forme de ruban…). La détermination des critères spécifiques permettant de détecter les régions correspondant aux surfaces sera aussi décrite dans le déliverable D13.



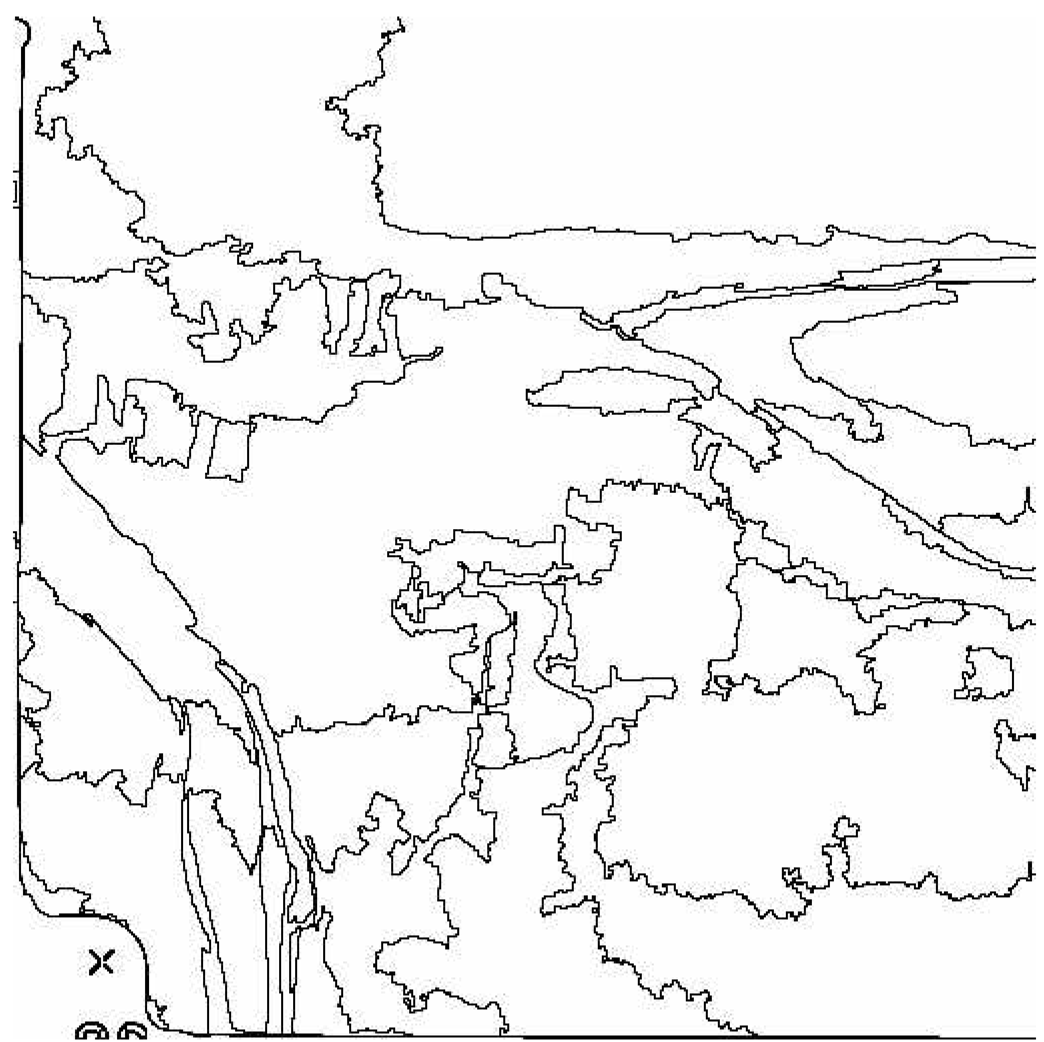
Résultat de la segmentation de l'image par la méthode de Mumford-Shah sur l'image 2



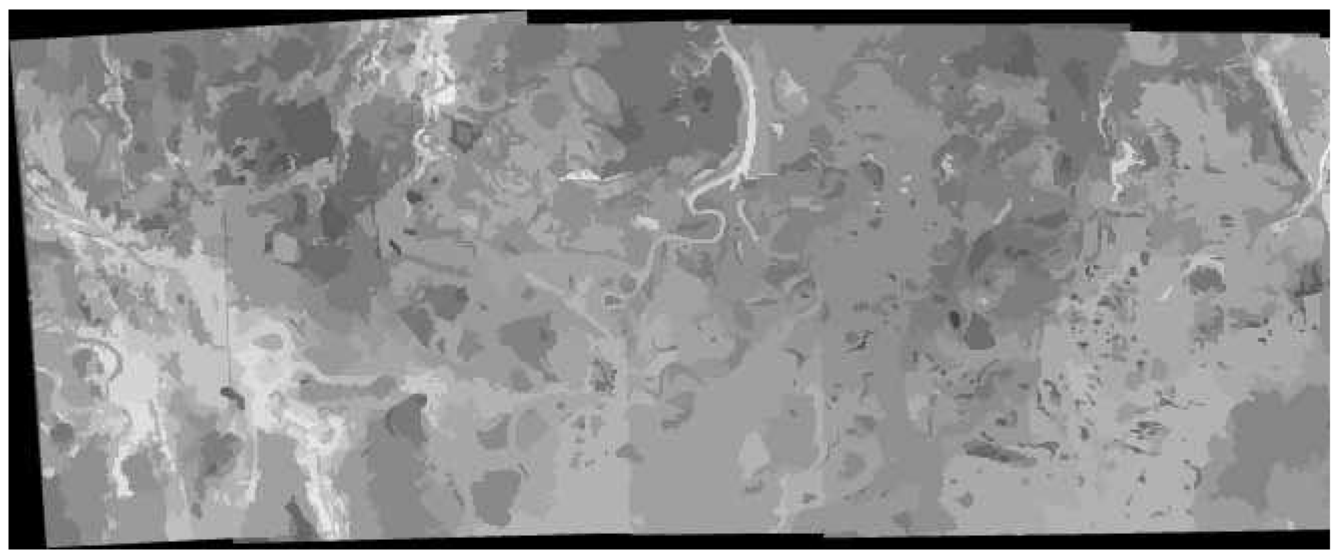
Zones inondés : image 7



Segmentation de l'image 7 par la méthode de Mimford-Shah



Contours des régions correspondant à l'image précédente



Segmentation de l'image 5 par la méthode de Mumford-Shah

5.3 Extraction de paturages secs et de ligneux

Les pâturages secs et les ligneux sont associés à des zones de texture homogènes de l'image. Afin de les détecter, nous mettons en œuvre un algorithme de segmentation d'images texturées

reposant sur la segmentation en zones homogènes de transformées de l'image. Dans un premier temps on calcule des transformées de l'image (module "mschannel") et ensuite on segmente en zones homogènes dans ce nouvel espace avec l'algorithme mis en œuvre précédemment. Une implémentation de cette méthode correspond au module "segtxt" de Megawave2 dont la description est dans l'extrait du manuel d'utilisation inclus ci dessus.

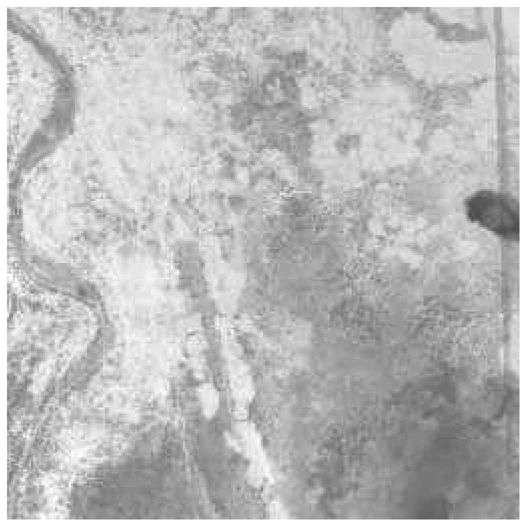
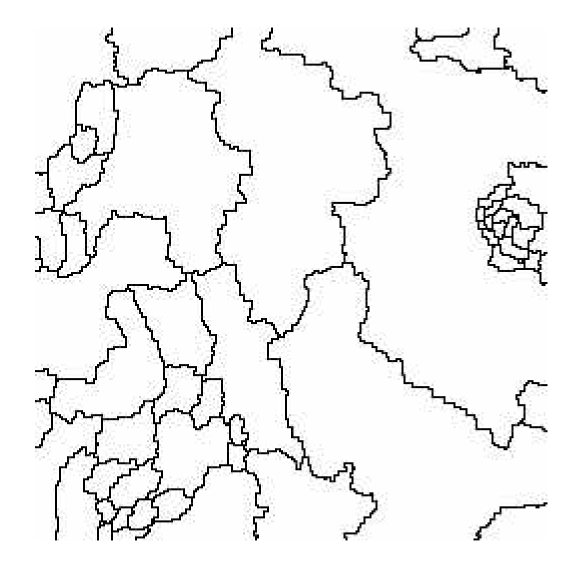
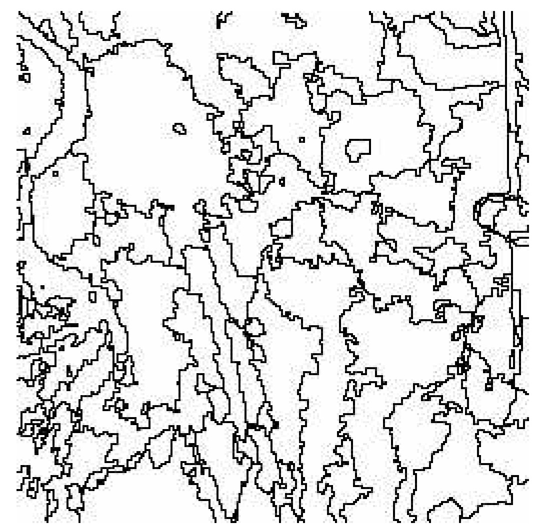


Image 8 : ligneux et paturages secs



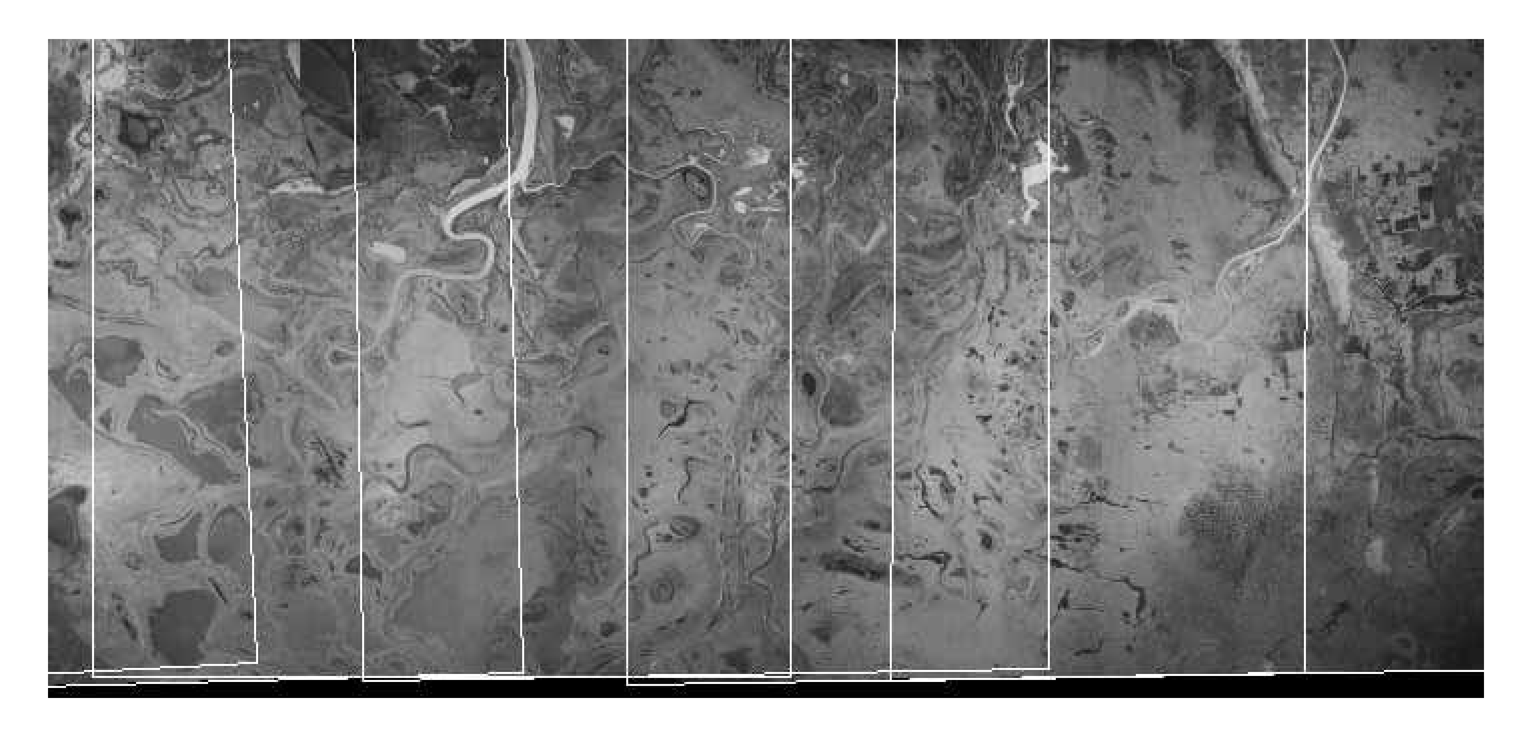
Contours des régions de texture homogène de l'image 8



Contours des régions homogènes (méthode de Mumford-Shah), les zones texturées ne sont pas détectées.

5.4 Mosaiquage des images

Cette partie correspond au déliverable D14 ("Registration of spatialized data"). Les algorithmes mis en œuvre s'appuie sur des modules de Targetjunior (détection de points caractéristiques par la méthode de Harris, mise en correspondance par des techniques robustes et calcul de la meilleure homographie, mosaiquage). Nous présentons juste ici un exemple de résultats, pour plus de détails on se référera au déliverable D14.



Résultat de mosaiquage d'images les contours des images initiales sont en blanc

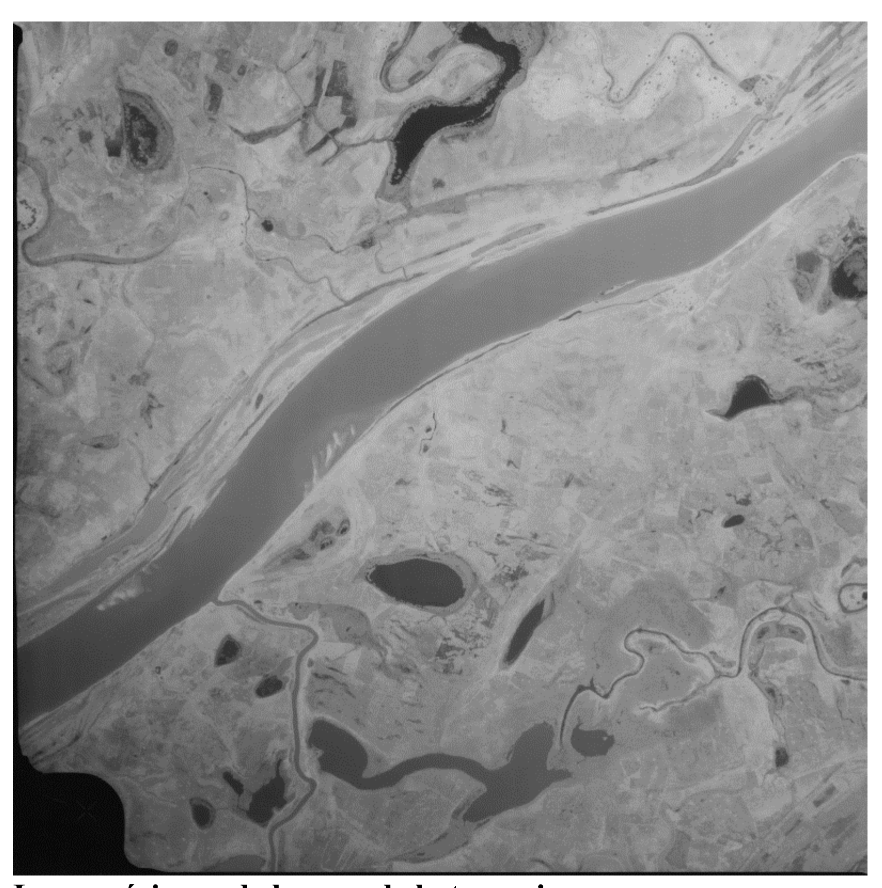
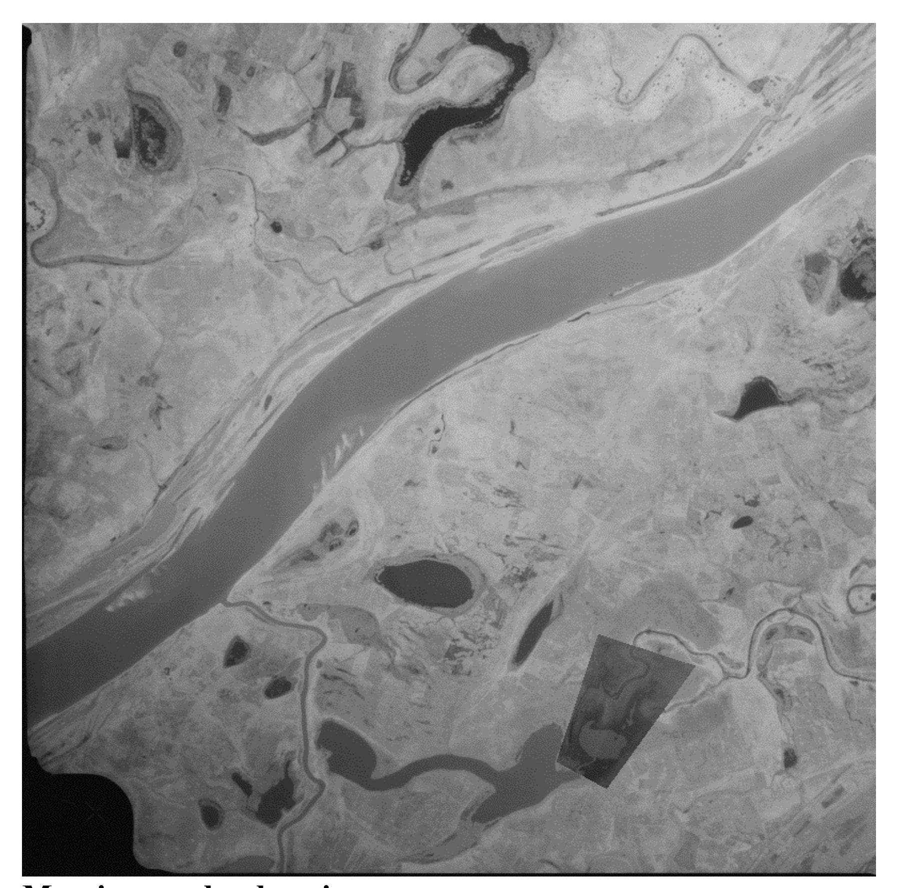


Image aérienne de la zone de batamani



Image aérienne prise à basse altitude de la mare de Batamani



Mosaiquage des deux images

5.5 Reconstruction de modèles numériques de terrain

Targetjunior contient les plus récents algorithmes de vision stéréoscopique avec ou sans calibration. On peut donc appliquer ces méthodes pour reconstruire des modèles numériques de terrain (MNT) à partir d'images aériennes (se recouvrant partiellement) non calibrés ou calibrés. Pour le moment, nous nous sommes principalement consacré à l'opération pilote Delta Central du Niger pour laquelle l'absence de relief ne rend pas très utile la reconstruction de MNT. Nous conservons donc cette possibilité pour d'autres opérations pilotes de SIMES à venir éventuellement d'ailleurs après la fin du contrat.